



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**

**ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ**

**ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

**Μελέτη και Σχεδιασμός Συστημάτων για τη Διαχείριση  
Υλικών με τη Χρήση Τεχνολογιών RFID**

**Study and Design of Systems for Item Management using  
RFID Technologies**



**ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ**

**του**

**Ελευθεριάδη Χαράλαμπου**

**Βόλος, Φεβρουάριος 2016**



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ  
ΥΠΟΛΟΓΙΣΤΩΝ

## **Μελέτη και Σχεδιασμός Συστημάτων για τη Διαχείριση Υλικών με τη Χρήση Τεχνολογιών RFID**

## **Study and Design of Systems for Item Management using RFID Technologies**

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ

του

**Ελευθεριάδη Χαράλαμπου**

**Επιβλέποντες :**

**ΤΣΟΜΠΙΑΝΟΠΟΥΛΟΥ ΠΑΝΑΓΙΩΤΑ**

Αναπληρώτρια Καθηγήτρια Π.Θ.

**ΜΠΟΖΑΝΗΣ ΠΑΝΑΓΙΩΤΗΣ**

Καθηγητής Π.Θ.

**ΒΑΣΙΛΑΚΟΠΟΥΛΟΣ ΜΙΧΑΗΛ**

Αναπληρωτής Καθηγητής Π.Θ.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την *ΗΜΕΡΟΜΗΝΙΑ ΕΞΕΤΑΣΗΣ*

(Υπογραφή)

.....  
ΤΣΟΜΠΙΑΝΟΠΟΥΛΟΥ ΠΑΝΑΓΙΩΤΑ  
Αναπληρώτρια Καθηγήτρια Π.Θ.

(Υπογραφή)

.....  
ΜΠΟΖΑΝΗΣ ΠΑΝΑΓΙΩΤΗΣ  
Καθηγητής Π.Θ.

(Υπογραφή)

.....  
ΒΑΣΙΛΑΚΟΠΟΥΛΟΣ ΜΙΧΑΗΛ  
Αναπληρωτής Καθηγητής Π.Θ.

Βόλος, Φεβρουάριος 2016

(Υπογραφή)

.....

**Ελευθεριάδης Χαράλαμπος**

Διπλωματική Εργασία για την απόκτηση Μεταπτυχιακού Διπλώματος Ειδίκευσης στην «Επιστήμη και Τεχνολογία Υπολογιστών, Τηλεπικοινωνιών και Δικτύων», στα Πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Θεσσαλίας

Copyright © Eleftheriadis Charalampos, 2016

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.



## Περίληψη

Στην παρούσα μεταπτυχιακή εργασία γίνεται μια προσπάθεια για την μελέτη και την περιγραφή της τεχνολογίας αναγνώρισης μέσω των ραδιοσυχνοτήτων (**RFID**: *Radio Frequency Identification*), μιας τεχνολογίας που τα τελευταία χρόνια εμφανίζεται συχνά στην καθημερινότητα του ανθρώπου και μπορεί να διευκολύνει σημαντικές πτυχές της ζωής του. Ο όρος RFID χρησιμοποιείται για να περιγράψει τα συστήματα εκείνα που εκπέμπουν και λαμβάνουν πληροφορία ασύρματα, με τη χρήση ραδιοκυμάτων. Η πληροφορία αυτή αποτελεί την ταυτότητα ενός συγκεκριμένου αντικειμένου ή ατόμου, σε μορφή ενός μοναδικού σειριακού αριθμού και είναι αποθηκευμένη σε ηλεκτρονικές ετικέτες (tags). Οι ετικέτες, που μπορεί να είναι προσκολλημένες ή ενσωματωμένες στα αντικείμενα, μεταδίδουν την πληροφορία που διαθέτουν ασύρματα στους αναγνώστες (readers), για ταυτοποίηση και για άλλους σκοπούς. Για να καταστεί εφικτή η μετάδοση, η επεξεργασία και η αποθήκευση της πληροφορίας στα συστήματα RFID, είναι απαραίτητη η χρήση συστατικών υλικού, λογισμικού, δικτύου και βάσεων δεδομένων. Μερικά συστήματα χρησιμοποιούν παθητικές ετικέτες χαμηλού κόστους με μικρή εμβέλεια ανάγνωσης και μικρό όγκο πληροφορίας (passive tags). Άλλα συστήματα χρησιμοποιούν υψηλών επιδόσεων ετικέτες που ενισχύονται από μπαταρία και μπορούν να διαθέτουν σημαντικό όγκο πληροφορίας και μεγάλη εμβέλεια ανάγνωσης (active tags). Τα τελευταία χρόνια παρατηρείται μια ραγδαία εξέλιξη στα συστήματα RFID, κάτι που έχει προκαλέσει το ενδιαφέρον και την προτίμηση πολλών επιχειρήσεων και οργανισμών παγκοσμίως, διευκολύνοντας πολλές από τις διαδικασίες τους με ταυτόχρονη μείωση κόστους και χρόνου που απαιτούνται για τις διαδικασίες αυτές. Στα πλαίσια της παρούσας εργασίας, σχεδιάζεται και υλοποιείται μια εφαρμογή που κάνει χρήση της τεχνολογίας RFID και εκμεταλλεύεται κάποια από τα στοιχεία που αναφέρθηκαν παραπάνω.



## **Abstract**

This thesis is an effort of studying and describing the technology of Radio Frequency Identification (RFID). This new technology has become a significant part of everyday life in recent years and is used to improve many aspects of human and business procedures. Radio Frequency Identification is a term that describes systems who transmit and receive information wirelessly, using radio waves. RFID enables this wireless data to be collected by readers from electronic tags attached to or embedded in objects, for identification and other purposes. RFID systems involve hardware, software, network and database components that enable information to flow from electronic tags to the infrastructure, where it is processed and stored. Some use passive, low-cost tags with short read ranges and only small amounts of information on tags. Others use active battery-powered, high-performance tags with high data capacity or read ranges that can carry considerable data on tags without network connection. In the last few years the great evolution of these systems attracted many companies and organizations worldwide in order to reduce the cost and time needed of many of their procedures and processes, improving business performance. In this thesis, an RFID technology-based application will be designed and implemented with the use of some of the above features.





# Περιεχόμενα

<b>1</b>	<b>Εισαγωγή .....</b>	<b>1</b>
1.1	Γενικά .....	1
1.2	Αντικείμενο μεταπτυχιακής διατριβής.....	1
1.3	Διάρθρωση της μεταπτυχιακής διατριβής.....	2
<b>2</b>	<b>Ιστορική αναδρομή .....</b>	<b>3</b>
2.1	Συστήματα-Πρόγονοι της τεχνολογίας RFID .....	3
2.2	Τα πρώτα RFID συστήματα .....	4
2.3	Η εξέλιξη της τεχνολογίας RFID.....	5
<b>3</b>	<b>Η τεχνολογία RFID.....</b>	<b>6</b>
3.1	Περιγραφή.....	6
3.1.1	Αυτόματη Αναγνώριση (Automatic Identification) .....	6
3.1.2	Αναγνώριση Μέσω Ραδιοσυχνοτήτων (Radio Frequency Identification).....	7
3.2	Σχεδιασμός συστημάτων RFID .....	7
3.2.1	Ετικέτες (Tags).....	7
3.2.2	Αναγνώστες (Readers).....	8
3.2.3	Συχνότητες RFID .....	9
3.3	Το πρωτόκολλο EPC Gen 2.....	10
3.4	Οι διάφορες χρήσεις της τεχνολογίας RFID .....	11
<b>4</b>	<b>Αλγόριθμοι εύρεσης συντομότερου μονοπατιού.....</b>	<b>12</b>
4.1	Εισαγωγή.....	12
4.2	Στατική δρομολόγηση .....	12
4.3	Αλγόριθμοι .....	13
4.3.1	Ο αλγόριθμος Dijkstra .....	13
4.3.2	Τροποποιημένος αλγόριθμος Dijkstra .....	14
4.3.3	Ο αλγόριθμος Bellman – Ford.....	15

4.3.4	Ευριστική αναζήτηση & ο αλγόριθμος A*	15
4.3.5	Αμφίδρομη αναζήτηση	17
4.3.6	Ιεραρχική αναζήτηση	18
4.4	Συμπεράσματα	19
5	Συστήματα πλοήγησης	20
5.1	Γενικά	20
5.2	Μέθοδοι εύρεσης τοποθεσίας	21
5.2.1	Καθολικό δορυφορικό σύστημα πλοήγησης - Global Navigation Satellite System (GNSS)	21
5.2.2	Υποβοηθούμενο καθολικό σύστημα εύρεσης τοποθεσίας – Assisted GPS	22
5.2.3	Κυψελοειδή δίκτυα – Cellular networks	22
5.2.3.1	Γωνία άφιξης (Angle of Arrival - AOA)	22
5.2.3.2	Γωνιακή διαφορά άφιξης (Angle Difference of Arrival)	22
5.2.3.3	Χρόνος άφιξης (Time of Arrival - TOA)	23
5.2.3.4	Χρονική διαφορά άφιξης (Time Difference of Arrival)	24
5.2.3.5	Η μέθοδος της «κυψέλης προέλευσης» - Cell-of-origin	25
5.2.3.6	Τριγωνισμός - Triangulation	25
5.2.3.7	Μέθοδος εύρεσης τοποθεσίας με τη χρήση αποτυπωμάτων – Location Fingerprinting	26
5.2.3.8	Σύνοψη	26
5.2.4	Ασύρματο τοπικό δίκτυο - WLAN (Wireless Local Area Network)	27
5.2.5	Bluetooth	27
5.2.6	Άλλες μέθοδοι εύρεσης τοποθεσίας	27
6	Αδρανειακό σύστημα πλοήγησης – Inertial Navigation System (INS)	29
6.1	Περιγραφή συστήματος	29
6.2	Dead Reckoning (DR)	29
6.3	Προσαρμογή στον χάρτη – Map Matching	30
6.4	Τεχνικές προσαρμογής	30
6.5	Γραφική αναπαράσταση	31
6.6	Αλγόριθμος δρομολόγησης	31
6.7	Συμπεράσματα	31

<b>7</b>	<b>Σχεδιασμός εφαρμογής.....</b>	<b>32</b>
7.1	Εισαγωγή.....	32
7.2	Περιγραφή συστήματος .....	33
7.2.1	Το μοντέλο «πελάτη – εξυπηρετητή» .....	33
7.2.2	Από τη μεριά του «πελάτη» (client – side).....	33
7.2.3	Από τη μεριά του «εξυπηρετητή» (server – side).....	34
7.3	Αρχιτεκτονική συστήματος.....	34
<b>8</b>	<b>Υλοποίηση.....</b>	<b>36</b>
8.1	Η φόρμα “Scanner” .....	36
8.2	Η φόρμα “New Product” .....	38
8.3	Η φόρμα “Add Product Info” .....	38
8.4	Η φόρμα “Product Information” .....	41
8.5	Η φόρμα “Edit Tag Data” .....	41
8.6	Η φόρμα “Shortest Path” .....	43
8.7	Η φόρμα “Navigation”.....	44
8.8	Η απομακρυσμένη βάση δεδομένων .....	45
8.9	Ενδεικτικός κώδικας υλοποίησης σε C#, SQL και PHP .....	46
<b>9</b>	<b>Επίλογος .....</b>	<b>49</b>
<b>10</b>	<b>Βιβλιογραφία - Πηγές .....</b>	<b>50</b>

**1.1 Γενικά**

Η τεχνολογία RFID (*Radio Frequency Identification*) είναι μια μορφή ασύρματης επικοινωνίας η οποία χρησιμοποιεί ραδιοκύματα, με σκοπό την αναγνώριση ή τον εντοπισμό ενός ή περισσοτέρων αντικειμένων. Εκμεταλλεύεται και εξελίσσει την ιδέα της τεχνολογίας ανάγνωσης barcode και την ψηφιοποιεί παρέχοντας τη δυνατότητα της γρήγορης, εύκολης και από απόσταση αναγνώρισης ενός προϊόντος. Η αναγνώριση αυτή μπορεί να γίνει χωρίς να υπάρχει άμεση οπτική επαφή με το εκάστοτε αντικείμενο, ενώ είναι εφικτή και η ταυτόχρονη αναγνώριση πολλαπλών αντικειμένων που μπορεί να απέχουν μεταξύ τους από λίγα εκατοστά έως μερικά μέτρα.

**1.2 Αντικείμενο μεταπτυχιακής διατριβής**

Με βάση την τεχνολογία RFID έγινε ο σχεδιασμός και η υλοποίηση μιας εφαρμογής που προορίζεται για PDA (*Personal Digital Assistant*), το οποίο θα χρησιμοποιηθεί για την ανάγνωση των ραδιοσυχνοτήτων. Η εφαρμογή θα είναι σε θέση να αναγνωρίζει συγκεκριμένα αντικείμενα-προϊόντα και να τα διαχειρίζεται με τη βοήθεια μιας βάσης δεδομένων στην οποία βρίσκονται αποθηκευμένες κάποιες πληροφορίες σχετικά με αυτά. Σε επόμενο στάδιο γίνεται έλεγχος για τη διαθεσιμότητα των αντικειμένων, ενώ είναι δυνατή και η ενημέρωση της βάσης δεδομένων με την εισαγωγή νέων αντικειμένων από τον χρήστη. Μπορεί να γίνει και αφαίρεση ενός προϊόντος από τη βάση δεδομένων σε περίπτωση που αυτό έχει λήξει, όπως επίσης και τροποποίηση της πληροφορίας (μοναδικό αναγνωριστικό) που φέρει η κάθε ετικέτα. Επιπροσθέτως, επιλέγεται και εμφανίζεται η συνιστώμενη συντομότερη διαδρομή για την εναπόθεση ή παραλαβή των αντικειμένων που εμπλέκονται στα διαφορετικά σενάρια εκτέλεσης. Στη συνέχεια γίνεται μια γενική περιγραφή των πιο διαδεδομένων αλγορίθμων εύρεσης συντομότερου μονοπατιού και παρουσιάζονται ορισμένα συμπεράσματα για τα αποτελέσματά τους και τον τρόπο χρήσης τους. Επίσης, γίνεται η μελέτη ορισμένων συστημάτων πλοήγησης που υπάρχουν και περιγράφονται οι λειτουργίες τους.

### **1.3 Διάρθρωση της μεταπτυχιακής διατριβής**

Το Κεφάλαιο 2 αποτελεί μια ιστορική αναδρομή στις τεχνολογίες και τα συστήματα που αποτέλεσαν τους προγόνους της τεχνολογίας αναγνώρισης μέσω των ραδιοσυχνοτήτων και παρουσιάζεται το πώς έγινε η μετάβαση στη μορφή που έχει σήμερα η τεχνολογία αυτή.

Στο Κεφάλαιο 3 γίνεται μια περιγραφή της τεχνολογίας RFID και των βασικών χαρακτηριστικών της. Παρουσιάζονται τα επιμέρους συστατικά που την αποτελούν και γίνεται ανάλυση των λειτουργιών τους. Έπειτα, αναφέρεται η σπουδαιότητα της τεχνολογίας αυτής και παρουσιάζονται οι διαφορετικοί τομείς της καθημερινότητας του ανθρώπου στους οποίους χρησιμοποιείται.

Στο Κεφάλαιο 4 γίνεται παρουσίαση των κυριότερων αλγορίθμων που χρησιμοποιούνται για την εύρεση συντομότερου μονοπατιού και μεθόδων περιήγησης.

Στο Κεφάλαιο 5 περιγράφονται τα διάφορα συστήματα πλοήγησης που υπάρχουν και οι σημαντικότερες μέθοδοι εύρεσης τοποθεσίας.

Στο Κεφάλαιο 6 παρουσιάζεται το «Αδρανειακό Σύστημα Πλοήγησης» (Inertial Navigation System) και οι μέθοδοι που χρησιμοποιούνται για τη δόμησή του.

Στο Κεφάλαιο 7 περιγράφεται η εφαρμογή που υλοποιείται στην παρούσα εργασία. Πιο συγκεκριμένα, αναφέρονται τα εργαλεία και οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για την ανάπτυξή της, ενώ παρουσιάζονται οι λειτουργίες της εφαρμογής καθώς και η αρχιτεκτονική του συστήματος που υλοποιήθηκε.

Στο Κεφάλαιο 8 παρουσιάζεται η υλοποίηση της εφαρμογής. Περιγράφονται οι λειτουργίες της και γίνεται αναφορά στα τμήματα και στις συναρτήσεις του κώδικα υλοποίησης που θεωρούνται σημαντικά.

Το Κεφάλαιο 9 αποτελεί τον επίλογο της εργασίας, με την παρουσίαση γενικών συμπερασμάτων ενώ στο Κεφάλαιο 10, βρίσκεται η βιβλιογραφία και οι πηγές που χρησιμοποιήθηκαν.

Στο Κεφάλαιο αυτό θα γίνει μια ιστορική αναδρομή στις τεχνολογίες και τα συστήματα που αποτέλεσαν τις ρίζες της τεχνολογίας RFID και θα παρουσιαστούν τα στάδια εξέλιξής της από τη στιγμή που εφαρμόστηκε για πρώτη φορά.

## **2.1 Συστήματα-Πρόγονοι της τεχνολογίας RFID**

Είναι κοινά αποδεκτό ότι οι ρίζες της τεχνολογίας RFID εντοπίζονται κατά τη διάρκεια του Δευτέρου Παγκοσμίου Πολέμου. Οι Γερμανοί, οι Ιάπωνες, οι Αμερικάνοι και οι Βρετανοί, χρησιμοποιούσαν συστήματα ραντάρ, τα οποία ανακαλύφθηκαν το 1935 από τον Σκωτσέζο φυσικό Sir Robert Alexander Watson-Watt. Τα συστήματα αυτά προειδοποιούσαν τα κέντρα ή της βάσης ελέγχου, όταν πλησίαζε κάποιο αεροπλάνο που βρισκόταν αρκετά χιλιόμετρα μακριά. Το πρόβλημα όμως ήταν ότι δεν υπήρχε τρόπος να αναγνωριστεί αν το αεροπλάνο αυτό ήταν εχθρικό ή όχι.

Οι Γερμανοί ανακάλυψαν ότι αν ένας πιλότος αναποδογύριζε το αεροπλάνο του κατά τον οριζόντιό του άξονα, θα άλλαζε την αντανάκλαση του ραδιοσήματος. Με τον τρόπο αυτό ειδοποιούνταν το πλήρωμα του ραντάρ ότι πρόκειται για Γερμανικό αεροπλάνο και όχι για αεροπλάνο των Συμμαχικών Δυνάμεων. Διαισθητικά αυτό ήταν και το πρώτο παθητικό σύστημα RFID.

Υπό την εποπτεία του Watson-Watt, ο οποίος διηύθυνε ένα απόρρητο πρόγραμμα, οι Βρετανοί ανέπτυξαν το πρώτο σύστημα «αναγνώρισης φίλου ή εχθρού» (IFF: Identify Friend or Foe). Τοποθέτησαν έναν πομπό σε κάθε Βρετανικό αεροπλάνο. Όταν ο πομπός λάμβανε σήματα από τους σταθμούς των ραντάρ στο έδαφος, άρχιζε να αναμεταδίδει ένα σήμα πίσω στους σταθμούς. Με το σήμα αυτό αναγνωριζόταν ότι το αεροπλάνο ήταν φιλικό.

Ένα RFID σύστημα, λειτουργεί με την ίδια βασική ιδέα: ένα σήμα στέλνεται σε έναν αναμεταδότη, ο οποίος με τη σειρά του «ξυπνάει» και είτε αντανakλάει ένα σήμα (passive system), είτε αναμεταδίδει ένα σήμα (active system).

Η ανάπτυξη και η εξέλιξη στις τεχνολογίες ραντάρ και στον τομέα των επικοινωνιών μέσω ραδιοσυχνοτήτων, συνεχίστηκε κατά τις δεκαετίες '50 και '60. Επιστήμονες και ακαδημαϊκοί στις ΗΠΑ, την Ευρώπη και την Ιαπωνία έκαναν έρευνες και παρουσίαζαν διατριβές για το πώς μπορούσε να γίνει χρήση της ραδιοσυχνότητας και της ενέργειάς της για την αναγνώριση/ταυτοποίηση αντικειμένων εξ' αποστάσεως. Κάποιες εταιρείες άρχισαν να διαφημίζουν αντικλεπτικά συστήματα που χρησιμοποιούσαν ραδιοκύματα για να χαρακτηρίσουν ένα αντικείμενο ως αγορασμένο ή όχι. Η τεχνολογία αυτή, που χρησιμοποιείται μέχρι και σήμερα, είναι γνωστή ως «Ηλεκτρονική Επιτήρηση Αντικειμένου» (EAS: Electronic Article Surveillance). Μια ετικέτα EAS διαθέτει ένα αναγνωριστικό μεγέθους 1 bit. Το bit είναι είτε ενεργό είτε ανενεργό. Αν κάποιος πελάτης πληρώσει για να αγοράσει το συγκεκριμένο αντικείμενο, το bit απενεργοποιείται και ο πελάτης μπορεί να εξέλθει του καταστήματος. Σε διαφορετική περίπτωση, αν ο πελάτης προσπαθήσει να βγει από το κατάστημα, οι αναγνώστες που είναι εγκατεστημένοι στις εισόδους ανιχνεύουν την ενεργοποιημένη ετικέτα και ηχούν συναγερμό.

## 2.2 Τα πρώτα RFID συστήματα

Ο Mario W. Cardullo, θεωρείται ότι σχεδίασε στις ΗΠΑ, την πρώτη ενεργή ετικέτα RFID (active RFID tag) με επανεγγράψιμη μνήμη, στις 23 Ιανουαρίου 1973. Την ίδια χρονιά ο Αμερικανός Charles Walton, λαμβάνει δίπλωμα ευρεσιτεχνίας για έναν παθητικό αναμεταδότη (passive transponder), η χρήση του οποίου έκανε μια πόρτα να ανοίγει χωρίς την ύπαρξη κλειδιού. Μια κάρτα με ενσωματωμένο αναμεταδότη επικοινωνούσε με έναν αναγνώστη (reader), που βρισκόταν κοντά στην πόρτα, στέλνοντάς του σήματα μέσω ραδιοκυμάτων. Όταν ο αναγνώστης εντόπιζε έναν έγκυρο αριθμό που ήταν αποθηκευμένος στην RFID ετικέτα, τότε ξεκλείδωνε την πόρτα.

Η κυβέρνηση των ΗΠΑ ενδιαφερόταν και αυτή για τα συστήματα RFID. Κατά τη δεκαετία του 1970, το Τμήμα Ενέργειας των ΗΠΑ, ζήτησε από το Εθνικό Εργαστήριο του Λος Άλαμος να αναπτύξει ένα σύστημα για τον εντοπισμό ραδιενεργών υλικών. Μία ομάδα επιστημόνων σκέφτηκε να εισάγει αναμεταδότες στα φορτηγά και αναγνώστες στις πύλες των εγκαταστάσεων ασφαλείας. Η κεραία στην πύλη θα «ξυπνούσε» τον αναμεταδότη στο φορτηγό, ο οποίος θα απαντούσε με ένα μοναδικό αναγνωριστικό (ID) και ενδεχομένως με κάποια επιπλέον δεδομένα (πχ. την ταυτότητα του οδηγού). Οι επιστήμονες που εργάστηκαν στο πρόγραμμα αυτό δημιούργησαν δική τους εταιρεία και στα μέσα της δεκαετίας του 1980 ανέπτυξαν ένα αυτοματοποιημένο σύστημα πληρωμής διοδίων. Τα συστήματα αυτά χρησιμοποιούνται παγκοσμίως μέχρι και σήμερα σε δρόμους, γέφυρες και τούνελ.

Το Εργαστήριο του Λος Άλαμος ανέλαβε επίσης την ανάπτυξη μιας παθητικής ετικέτας RFID (passive RFID tag) που χρησιμοποιούσε ραδιοκύματα σε συχνότητες UHF (Ultra High Frequency: 300MHz – 3GHz), ύστερα από αίτημα του Τμήματος Γεωργίας των ΗΠΑ, για τον εντοπισμό αγελάδων. Το πρόβλημα ήταν ότι στις αγελάδες χορηγούσαν ορμόνες και φάρμακα όταν αυτές αρρώσταιναν. Ήταν όμως δύσκολο να σιγουρευτούν οι κτηνοτρόφοι ότι σε κάθε αγελάδα δόθηκε η σωστή ποσότητα και όχι διπλή δόση κατά λάθος. Μερικά χρόνια αργότερα, άλλες εταιρείες ανέπτυξαν συστήματα χαμηλής συχνότητας (125 kHz), που εμπεριείχαν μικρότερους σε μέγεθος αναμεταδότες. Ένας αναμεταδότης, που ήταν κλεισμένος μέσα σε μικροσκοπική γυάλινη κάψουλα, τοποθετούνταν κάτω από το δέρμα της αγελάδας.



Οι αναμεταδότες χαμηλής συχνότητας τοποθετήθηκαν επίσης σε κάρτες και χρησιμοποιούνταν για τον έλεγχο εισόδου σε κτήρια.

## 2.3 Η εξέλιξη της τεχνολογίας RFID

Με το πέρασμα των χρόνων πολλές εταιρείες εμπορευματοποίησαν τα συστήματα χαμηλής συχνότητας (125 kHz) και έπειτα άρχισαν να κάνουν χρήση του φάσματος υψηλών συχνοτήτων (13.56 MHz), το οποίο ήταν αχρησιμοποίητο στα περισσότερα μέρη του κόσμου. Η υψηλή συχνότητα προσφέρει κάλυψη μεγαλύτερης απόστασης και γρηγορότερη μεταφορά δεδομένων. Σήμερα, τα RFID συστήματα των 13.56 MHz χρησιμοποιούνται σε ελέγχους πρόσβασης, σε συστήματα πληρωμών, σε «έξυπνες» κάρτες που δεν απαιτούν άμεση επαφή με συσκευή και σαν αντικλεπτικές συσκευές αυτοκινήτων.

Στις αρχές της δεκαετίας του 1990, οι μηχανικοί της IBM ανέπτυξαν ένα σύστημα RFID σε συχνότητες UHF. Οι συχνότητες αυτές, παρέχουν μεγαλύτερη απόσταση ανάγνωσης που φτάνει μέχρι και τα 6 μέτρα με ευνοϊκές συνθήκες, όπως επίσης και γρηγορότερη μεταφορά δεδομένων. Η UHF RFID τεχνολογία ενισχύθηκε σημαντικά το 1999, όταν δύο καθηγητές από το Ινστιτούτο Τεχνολογίας της Μασαχουσέτης (MIT), ο David Brock και ο Sanjay Sarma, εξέτασαν το ενδεχόμενο της τοποθέτησης ετικετών RFID χαμηλού κόστους σε όλα τα προϊόντα, με σκοπό τον εντοπισμό τους σε οποιοδήποτε σημείο της εφοδιαστικής αλυσίδας. Η ετικέτα θα διέθετε μόνο έναν σειριακό αριθμό με σκοπό τη μείωση τους κόστους της. Τα δεδομένα που συνδέονταν με τον σειριακό αριθμό της ετικέτας θα αποθηκεύονταν σε μια βάση δεδομένων που θα ήταν προσπελάσιμη μέσω του διαδικτύου.

Οι Sarma και Brock, ουσιαστικά άλλαξαν τον τρόπο που χρησιμοποιούνταν η τεχνολογία RFID στην εφοδιαστική αλυσίδα. Προηγουμένως, οι ετικέτες αποτελούσαν μια κινητή βάση δεδομένων που μετέφερε πληροφορίες για το προϊόν το οποίο συνόδευε. Οι Sarma και Brock εξέλιξαν την τεχνολογία αυτή σε μια τεχνολογία δικτύου, με τη διασύνδεση του αντικειμένου στο διαδίκτυο, διαμέσου της ετικέτας. Για τις επιχειρήσεις, το γεγονός αυτό αποτέλεσε μια μεγάλη και ιδιαίτερα σημαντική αλλαγή, διότι πλέον ο κατασκευαστής μπορούσε να ενημερώσει αυτόματα έναν επαγγελματικό συνεργάτη για το πότε ένα φορτίο έφευγε από τις εγκαταστάσεις του ή την αποθήκη. Με τη σειρά του ο έμπορος λιανικής μπορούσε να ενημερώσει αυτόματα τον κατασκευαστή πότε και ποια προϊόντα παρέλαβε. [3]

Στο Κεφάλαιο 3 θα γίνει μια περιγραφή της τεχνολογίας αυτόματης αναγνώρισης και ειδικότερα της τεχνολογίας RFID. Παρουσιάζονται τα συστατικά που αποτελούν ένα RFID σύστημα και περιγράφεται ο τρόπος λειτουργίας τους. Στη συνέχεια αναφέρονται οι συχνότητες λειτουργίας των συστημάτων αυτών, τα πρότυπα και τα πρωτόκολλα που ακολουθούν, δίνοντας έμφαση στο πρωτόκολλο EPC Gen 2, το οποίο και χρησιμοποιείται στην συγκεκριμένη εργασία. Τέλος, αναφέρονται οι διάφορες χρήσεις της τεχνολογίας RFID στην καθημερινότητα του ανθρώπου.

### **3.1 Περιγραφή**

#### **3.1.1 Αυτόματη Αναγνώριση (Automatic Identification)**

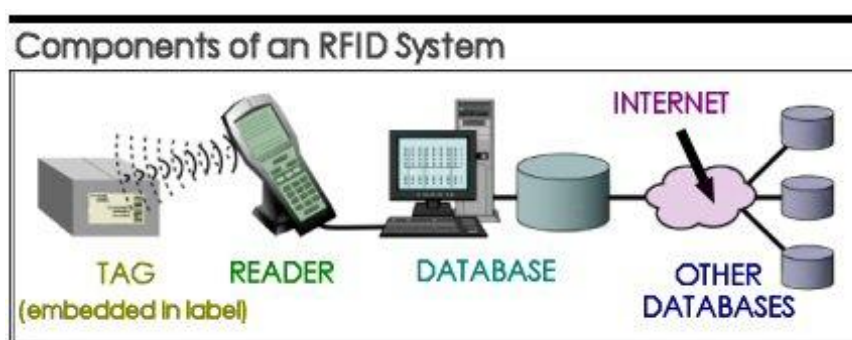
Ο όρος «Αυτόματη Αναγνώριση» (Automatic Identification) χρησιμοποιείται για να περιγράψει όλες εκείνες τις τεχνολογίες που έχουν στόχο να βοηθήσουν τα μηχανήματα στο να αναγνωρίζουν αντικείμενα. Η Αυτόματη Αναγνώριση συχνά συνδυάζεται με την αυτόματη συλλογή δεδομένων. Κι αυτό γιατί οι εταιρείες επιθυμούν να αναγνωρίζουν αντικείμενα, να αντλούν πληροφορίες σχετικά με αυτά και με κάποιον τρόπο να μεταφέρουν τα δεδομένα σε έναν ηλεκτρονικό υπολογιστή χωρίς να είναι αναγκαία η ύπαρξη κάποιου υπαλλήλου για τη διαδικασία αυτή. Σκοπός των περισσότερων συστημάτων αυτόματης αναγνώρισης είναι η αύξηση της αποδοτικότητας, η μείωση των σφαλμάτων κατά τη διαδικασία εισαγωγής δεδομένων και η καλύτερη αξιοποίηση του εργασιακού προσωπικού σε πιο εποικοδομητικές εργασίες. Οι τεχνολογίες που ανήκουν στην κατηγορία της Αυτόματης Αναγνώρισης είναι: bar codes, «έξυπνες κάρτες» (smart cards), αναγνώριση φωνής (voice recognition), οπτική αναγνώριση χαρακτήρων (optical character recognition) και η αναγνώριση μέσω ραδιοσυχνοτήτων.

### 3.1.2 Αναγνώριση Μέσω Ραδιοσυχνοτήτων (Radio Frequency Identification)

Ο όρος «Αναγνώριση Μέσω Ραδιοσυχνοτήτων» (Radio Frequency Identification ή RFID), αναφέρεται στις τεχνολογίες εκείνες που κάνουν χρήση ραδιοκυμάτων για την αναγνώριση ανθρώπων ή αντικειμένων. Υπάρχουν διάφοροι μέθοδοι αναγνώρισης, αλλά η πιο διαδεδομένη είναι η αποθήκευση ενός σειριακού αριθμού, ενδεχομένως και άλλων πληροφοριών, σε ένα μικροτσίπ το οποίο είναι προσκολλημένο σε μια κεραία. Το τσιπ αυτό σε συνδυασμό με την κεραία, αποτελούν τον αναμεταδότη RFID (RFID transponder) ή αλλιώς την ετικέτα RFID (RFID tag). Η κεραία ενεργοποιεί το μικροτσίπ να εκπέμψει την πληροφορία αναγνώρισης σε έναν αναγνώστη (reader).[1][5]

Στη συνέχεια, ο αναγνώστης μετατρέπει τα ραδιοκύματα που δέχεται από την ετικέτα RFID σε ψηφιακή πληροφορία η οποία μπορεί εισαχθεί σε υπολογιστές για περαιτέρω χρήση και επεξεργασία.

Ένα σύστημα RFID εμφανίζεται στην Εικόνα 3.1 που ακολουθεί:



Εικόνα 3.1- Συστατικά ενός συστήματος RFID

## 3.2 Σχεδιασμός συστημάτων RFID

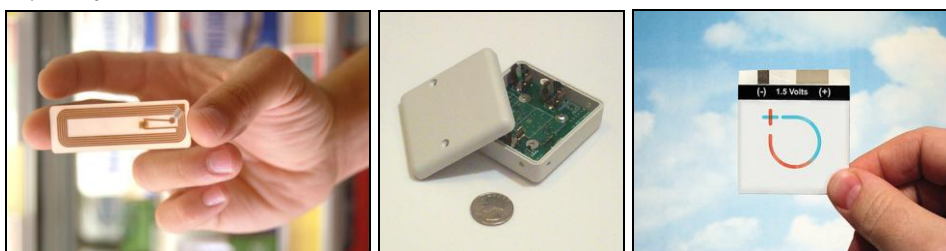
### 3.2.1 Ετικέτες (Tags)

Ένα σύστημα RFID χρησιμοποιεί ετικέτες που είναι προσκολλημένες ή ενσωματωμένες στο αντικείμενο προς αναγνώριση. Έπειτα ένας πομποδέκτης ραδιοκυμάτων, που ονομάζεται αναγνώστης (reader), στέλνει ένα σήμα στην ετικέτα και λαμβάνει/διαβάζει την «απάντησή» της.

Μια ετικέτα RFID μπορεί να είναι παθητική (passive), ενεργή (active), ή παθητική με υποβοήθηση μπαταρίας (battery-assisted passive). Η ενεργή ετικέτα διαθέτει ενσωματωμένη μια μπαταρία και περιοδικά εκπέμπει το αναγνωριστικό της σήμα. Η παθητική ετικέτα με υποβοήθηση μπαταρίας έχει ενσωματωμένη μια μικρή μπαταρία και ενεργοποιείται μόνο όταν παρουσιαστεί κάποιος αναγνώστης στην εμβέλειά της. Η παθητική ετικέτα είναι η φθηνότερη και η μικρότερη σε μέγεθος, καθώς δε διαθέτει μπαταρία. Αντ' αυτού, η ετικέτα χρησιμοποιεί την ενέργεια από τα ραδιοκύματα που εκπέμπει ο αναγνώστης.

Οι ετικέτες μπορεί να προορίζονται είτε μόνο για ανάγνωση (read-only), είτε και για ανάγνωση και για εγγραφή (read/write). Οι ετικέτες που είναι μόνο για ανάγνωση, έχουν προεγκατεστημένο από τον κατασκευαστή έναν σειριακό αριθμό που χρησιμοποιείται ως «κλειδί» σε μια βάση δεδομένων. Οι ετικέτες που είναι και για ανάγνωση και για εγγραφή, μπορούν να δεχθούν τροποποίηση των δεδομένων τους από τον διαχειριστή του RFID συστήματος.

Οι ετικέτες RFID αποτελούνται από τουλάχιστον δύο μέρη: ένα ολοκληρωμένο κύκλωμα (integrated circuit) και μια κεραία (antenna). Το ολοκληρωμένο κύκλωμα χρησιμοποιείται για αποθήκευση και επεξεργασία πληροφορίας, για προσαρμογή του σήματος ραδιοσυχνότητας, για συλλογή της DC ισχύος από το σήμα του αναγνώστη και άλλες εξειδικευμένες λειτουργίες. Η κεραία χρησιμοποιείται για την λήψη και την εκπομπή του σήματος. Στην Εικόνα 3.2 παρουσιάζονται τα είδη των ετικετών που αναφέρθηκαν προηγουμένως:



Εικόνα 3.2 – Passive Tag (αριστερά), Active Tag (κέντρο), Battery-Assisted Passive Tag (δεξιά)

### 3.2.2 Αναγνώστες (Readers)

Ένας αναγνώστης RFID εκπέμπει ένα κωδικοποιημένο σήμα μέσω ραδιοσυχνότητας με σκοπό την αναγνώριση της ετικέτας RFID. Η ετικέτα με τη σειρά της, λαμβάνει το μήνυμα και απαντάει στέλνοντας την ταυτοποίηση και πιθανώς επιπλέον πληροφορίες. Η απάντηση αυτή μπορεί να περιλαμβάνει μόνο έναν μοναδικό σειριακό αριθμό, ή και πληροφορίες που σχετίζονται με το αντικείμενο, όπως κωδικός προϊόντος, όνομα, αριθμός συνόλου παραγωγής, ημερομηνία παρασκευής/λήξης, κτλ. Το γεγονός ότι κάθε ετικέτα διαθέτει τον δικό της μοναδικό αριθμό, κάνει τους αναγνώστες να μπορούν να αναγνωρίσουν πολλαπλές ετικέτες ταυτόχρονα, όταν αυτές βρεθούν στην εμβέλειά τους.

Οι αναγνώστες διαχωρίζονται σε σταθερούς και κινητούς (Εικόνα 3.3). Οι σταθεροί αναγνώστες μπορούν να εγκατασταθούν σε συγκεκριμένα σταθερά σημεία, ούτως ώστε να δημιουργείται μια ελεγχόμενη ζώνη ανάγνωσης ετικετών. Οι κινητοί αναγνώστες, μπορεί να είναι χειρός, επομένως εύκολα μεταφέρσιμοι, ή να εγκαθίστανται σε αυτοκίνητα και άλλα οχήματα.



Εικόνα 3.3 - Σταθερός RFID Reader και κεραία(αριστερά), RFID Reader χειρός (δεξιά)

### 3.2.3 Συχνότητες RFID

Τα συστήματα RFID μπορούν να λειτουργήσουν σε μια από τις ακόλουθες συχνότητες: χαμηλή συχνότητα (LF – Low Frequency), υψηλή συχνότητα (HF – High Frequency) και πολύ υψηλή συχνότητα (UHF – Ultra High Frequency). Τα ραδιοκύματα συμπεριφέρονται διαφορετικά σε κάθε μια από αυτές τις συχνότητες παρουσιάζοντας κάποια πλεονεκτήματα και μειονεκτήματα.[4]

- **Χαμηλή Συχνότητα (LF)**

Η ζώνη χαμηλών συχνοτήτων καλύπτει συχνότητες που κυμαίνονται από 30 KHz έως 300 KHz. Τα LF RFID συστήματα λειτουργούν συνήθως στα 125 KHz, αν και υπάρχουν μερικά συστήματα που λειτουργούν στα 134 KHz. Αυτή η ζώνη συχνοτήτων παρέχει μικρή εμβέλεια ανάγνωσης, της τάξης των 10 cm, και έχει μικρότερη ταχύτητα ανάγνωσης από τις υψηλές συχνότητες. Ωστόσο, η ζώνη χαμηλών συχνοτήτων συμπεριφέρεται καλύτερα σε παρεμβολή εμποδίων ανάμεσα στην ετικέτα και τον αναγνώστη. Στις εφαρμογές που χρησιμοποιούν τις χαμηλές συχνότητες, περιλαμβάνονται εφαρμογές ελέγχου πρόσβασης και εντοπισμός ζώων. Τα πρότυπα για την αναγνώριση στις χαμηλές συχνότητες, ορίζονται στις πιστοποιήσεις ISO 14223 και ISO/IEC 18000-2. Το κόστος των LF ετικετών κυμαίνεται μεταξύ 50 cents και \$2.

- **Υψηλή Συχνότητα (HF)**

Η ζώνη υψηλών συχνοτήτων κυμαίνεται από 3 MHz έως 30 MHz. Τα πιο πολλά HF RFID συστήματα λειτουργούν στα 13.56 MHz με εμβέλειες ανάγνωσης από 10 cm έως 1 μέτρο. Τα HF συστήματα είναι αρκετά ευαίσθητα στην παρεμβολή εμποδίων, ενώ παρουσιάζουν πιο γρήγορους ρυθμούς μεταφοράς δεδομένων. Τα HF RFID συστήματα χρησιμοποιούνται κυρίως στις πληρωμές εισιτηρίων και σε εφαρμογές μεταφοράς δεδομένων. Υπάρχουν αρκετά πρότυπα αναγνώρισης σε υψηλές συχνότητες. Ένα από αυτά είναι το ISO 15693 για τον εντοπισμό αντικειμένων και τα ECMA-340 και ISO/IEC 18092 για την τεχνολογία NFC (Near Field Communication). Άλλα πρότυπα υψηλών συχνοτήτων περιλαμβάνουν τα ISO/IEC 14443, ISO/IEC 14443 A και JIS X 6319-4, τα οποία χρησιμοποιούνται σε «έξυπνες κάρτες» (smart cards) και σε ηλεκτρονικές κάρτες συναλλαγών. Το κόστος των HF ετικετών κυμαίνεται μεταξύ 50 cents και \$2.

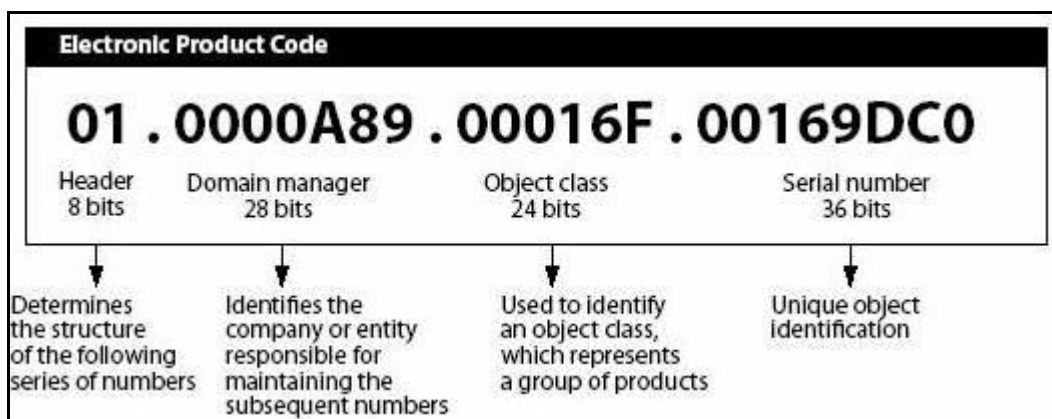
- **Πολύ Υψηλή Συχνότητα (UHF)**

Η UHF ζώνη συχνοτήτων κυμαίνεται από 300 MHz έως 3 GHz. Τα συστήματα που ακολουθούν το πρότυπο UHF Gen2 για RFID, χρησιμοποιούν τη ζώνη μεταξύ 860 και 960 MHz. Τα UHF Gen2 RFID συστήματα στις πιο πολλές χώρες λειτουργούν μεταξύ 900 και 915 MHz. Η εμβέλεια ανάγνωσης των UHF συστημάτων μπορεί να φτάσει μέχρι και τα 12 μέτρα, ενώ τα συστήματα αυτά διαθέτουν γρηγορότερους ρυθμούς μεταφοράς δεδομένων από τα LF και HF συστήματα. Επίσης, είναι τα πιο ευαίσθητα στην παρεμβολή εμποδίων, ωστόσο οι κατασκευαστές των UHF RFID συστημάτων έχουν ανακαλύψει τρόπους σχεδιασμού ετικετών, κεραιών και αναγνώστών που διατηρούν τις επιδόσεις σε υψηλά επίπεδα ακόμα και όταν οι συνθήκες σε ένα περιβάλλον είναι δύσκολες. Οι πολύ υψηλές συχνότητες χρησιμοποιούνται σε εφαρμογές που σχετίζονται με την καταγραφή εμπορευμάτων λιανικού εμπορίου, την ασύρματη ρύθμιση συσκευών, την αποφυγή παραποίησης φαρμάκων, κτλ. Το πρότυπο EPCglobal Gen2, είναι το μοναδικό πρότυπο για τις UHF συχνότητες, με πιστοποίηση ISO 18000-6C. Το πρότυπο EPC Gen2 χρησιμοποιείται στην εφαρμογή της παρούσας διπλωματικής και περιγράφεται παρακάτω. Το κόστος των UHF ετικετών κυμαίνεται μεταξύ 5 και 15 cents.

### 3.3 Το πρωτόκολλο EPC Gen 2

Ο όρος EPC Gen 2 είναι συντομογραφία για τον τίτλο EPCglobal UHF Class 1 Generation2. Η συντομογραφία EPC περιγράφει τον Ηλεκτρονικό Κωδικό Προϊόντος: Electronic Product Code (Εικόνα 3.4). Το EPC αντικατέστησε τον παγκόσμιο κωδικό προϊόντος Universal Product Code (UPC) που βρισκόταν σε πολλές ετικέτες αντικειμένων και αποτελούνταν από ένα σύνολο αριθμών σε συνδυασμό με ένα barcode. Ουσιαστικά το EPC αποτελεί το ισοδύναμο του UPC για την σημερινή αναγνώριση προϊόντων με τη χρήση της τεχνολογίας RFID. Ο Ηλεκτρονικός Κωδικός Προϊόντος αποτελείται από 96 bits και διαχωρίζεται σε τέσσερα διακριτά πεδία:

- **Επικεφαλίδα (Header):** 8 bits που προσδιορίζουν τη δομή των ακολουθούμενων αριθμών του EPC
- **Διαχειριστής EPC (Domain Manager):** 28 bits που προσδιορίζουν την εταιρεία ή την οντότητα που διαχειρίζεται τους ακολουθούμενους αριθμούς. Συχνά, το πεδίο αυτό, αναφέρεται στον κατασκευαστή του προϊόντος.
- **Κλάση του Αντικειμένου (Object Class):** 24 bits που χρησιμοποιούνται για την αναγνώριση μιας κλάσης αντικειμένων, η οποία αποτελεί ένα σύνολο από προϊόντα.
- **Σειριακός Αριθμός (Serial Number):** 36 bits που αποτελούν το μοναδικό αναγνωριστικό που προσδιορίζει το αντικείμενο



Εικόνα 3.4 - Electronic Product Code (EPC)

Η EPCglobal είναι μια συλλογική δράση που αποτελείται από παγκόσμιους, μη κερδοσκοπικούς οργανισμούς (GS1) και έχει σαν στόχο την ανάπτυξη και τη συντήρηση προτύπων για τις αλυσίδες εφοδιασμού και ζήτησης. Η GS1 συνεργάζεται με οργανισμούς βιομηχανίας, κυβερνήσεις και παρόχους τεχνολογίας και αναλαμβάνει να καλύψει τις ανάγκες τους με την υιοθέτηση και την υλοποίηση παγκοσμίων προτύπων.

Μία από τις αποστολές της EPCglobal ήταν να απλοποιήσει τα πρωτόκολλα που επικρατούσαν στις τεχνολογίες RFID κατά τη δεκαετία του 1990. Έτσι, το 2003, ορίστηκαν δύο «διεπαφές αέρα» (tag air interfaces), που αποτέλεσαν τα πρωτόκολλα για ανταλλαγή πληροφορίας ανάμεσα σε μια ετικέτα και σε έναν αναγνώστη. Τα πρωτόκολλα αυτά, που έγιναν γνωστά ως Class 0 και Class 1, γνώρισαν μεγάλη εμπορική εφαρμογή από το 2002 έως το 2005.

Το πρότυπο EPC Gen 2, έγινε δεκτό και εφαρμόστηκε τον Δεκέμβριο του 2004 και με κάποιες μικρές τροποποιήσεις, υιοθετήθηκε ως ISO 18000-6C το 2006. Το EPC Gen 2 ορίζει τις φυσικές και λογικές απαιτήσεις που χρειάζεται ένα RFID σύστημα (RFID Gen 2 Reader, EPC global tags) για να λειτουργήσει στο εύρος συχνοτήτων 860 MHz ~ 960 MHz.[6]

### **3.4    *Οι διάφορες χρήσεις της τεχνολογίας RFID***

Η τεχνολογία RFID έχει ενσωματωθεί και εφαρμόζεται σε πολλούς τομείς της καθημερινής ζωής του ανθρώπου. Μερικοί από αυτούς είναι:

- Έλεγχος πρόσβασης σε κτήρια ή εγκαταστάσεις
- Εντοπισμός και αναγνώριση αντικειμένων
- Εντοπισμός και αναγνώριση ατόμων και ζώων
- Πληρωμή διοδίων και πληρωμές χωρίς άμεση επαφή συσκευών/καρτών
- Τομέας υγείας (νοσοκομεία, φαρμακοβιομηχανίες, κτλ.)
- Τομέας εκπαίδευσης (σχολεία, πανεπιστήμια, μουσεία, κτλ.) [2]
- Αθλητισμός και σπορ

## 4

## Αλγόριθμοι εύρεσης

### συντομότερου

### μονοπατιού

Στο Κεφάλαιο αυτό θα γίνει η περιγραφή και η μελέτη των αλγορίθμων που χρησιμοποιούνται για την εύρεση του συντομότερου μονοπατιού σε ένα δίκτυο από κόμβους και ακμές.

#### 4.1 Εισαγωγή

Οι αλγόριθμοι και οι μέθοδοι για τον προγραμματισμό μιας διαδρομής σε δίκτυα μεταφορών έχουν υποστεί ραγδαία ανάπτυξη τα τελευταία χρόνια, γεγονός που οδήγησε στην δημιουργία μεθόδων οι οποίες είναι μέχρι και ένα εκατομμύριο γρηγορότερες από τον αλγόριθμο Dijkstra που υλοποιήθηκε στην παρούσα διπλωματική.

#### 4.2 Στατική δρομολόγηση

Θεωρείται ένα κατευθυνόμενο γράφημα  $G = (V, E)$  με  $n$  κόμβους και  $m = \Theta(n)$  ακμές. Μία ακμή  $(u, v)$  διαθέτει το μη αρνητικό βάρος  $w(u, v)$ . Σκοπός είναι να βρεθεί το ελάχιστο βάρος  $d(s, t)$  για οποιοδήποτε μονοπάτι μεταξύ ενός αρχικού κόμβου  $s$  και ενός κόμβου προορισμού  $t$ . Στην στατική δρομολόγηση, τα βάρη των ακμών παραμένουν σταθερά, πράγμα που είναι λογικό προκειμένου να γίνουν κάποιοι αρχικοί υπολογισμοί, να αποθηκευτούν τα αποτελέσματά τους και να γίνει χρήση της πληροφορίας αυτής για να επιταχυνθεί η διαδικασία αναζήτησης. [10][11]



## 4.3 Αλγόριθμοι

### 4.3.1 Ο αλγόριθμος Dijkstra

Ο κλασικός αλγόριθμος για τον σχεδιασμό δρομολόγησης ο οποίος υλοποιήθηκε και σε αυτήν την εργασία. Θεωρείται ένα γράφημα  $G(V,E)$ , όπου  $V$  το σύνολο των κόμβων του και  $E$  το σύνολο των ακμών του. Επίσης, θεωρείται μια συνάρτηση βάρους  $w : E \rightarrow \mathbb{R}^+ \cup \{0\}$  ορισμένη στις ακμές του γράφου. Αυτό σημαίνει ότι για την μετάβαση από έναν κόμβο του γράφου σε έναν άλλο, θα υπάρχει κάποιο κόστος. Αν τα βάρη είναι αρνητικά ο αλγόριθμος επιστρέφει λανθασμένο αποτέλεσμα.

Ο αλγόριθμος του Dijkstra βρίσκει τα μονοπάτια που πρέπει να ακολουθηθούν από έναν κόμβο-αφετηρία προς τους υπόλοιπους, ώστε να επιτευχθεί το λιγότερο δυνατό κόστος. Για τη λειτουργία του αλγόριθμου, σε ένα διάνυσμα  $d[]$  μεγέθους  $|V| = n$  αποθηκεύεται η μέχρι τώρα υπολογισμένη απόσταση των κόμβων από την αφετηρία. Κατά την αρχικοποίηση, οι αποστάσεις σημειώνονται  $d[s] = 0$  και  $d[v] = +\infty$  για κάθε  $v \neq s$ , όπου  $s$  είναι ο κόμβος-αφετηρία. Επιπλέον ο αλγόριθμος διατηρεί μια ουρά προτεραιότητας  $Q$ , για την επεξεργασία των κόμβων του γραφήματος στη σωστή σειρά, και ένα σύνολο  $S$ , το σύνολο των κόμβων για τους οποίους ο αλγόριθμος έχει βρει την ελάχιστη διαδρομή. Στην  $Q$  εισάγονται όλοι οι κόμβοι του γραφήματος με κλειδί την τιμή  $d[*]$ , ενώ το σύνολο  $S$  είναι αρχικά κενό. Τέλος, ο αλγόριθμος χρησιμοποιεί ένα ακόμη διάνυσμα, το  $prev[]$ , μεγέθους  $n$ , στο οποίο για κάθε κόμβο  $u$  αποθηκεύεται ο αμέσως προηγούμενος κόμβος στο ελάχιστο μονοπάτι προς τον  $u$ . Μετά την αρχικοποίηση των δομών που χρησιμοποιεί, ο αλγόριθμος εξάγει από την  $Q$  τον κόμβο  $x$  με το ελάχιστο  $d[*]$  και τον εισάγει στο σύνολο  $S$ . Στο πρώτο βήμα, για παράδειγμα, θα εξάγει τον κόμβο-αφετηρία  $s$ , αφού  $d[s]=0$  ενώ όλοι οι υπόλοιποι κόμβοι έχουν άπειρο  $d[*]$ . Για κάθε γείτονα  $y$  (του  $x$ ) που δεν ανήκει στο σύνολο  $S$ , αν  $d[y] > d[x] + w(x,y)$  τότε ενημερώνει το  $d[y]$  καταχωρώντας του την τιμή  $d[x] + w(x,y)$  και θέτει  $prev[y]=x$ . Δηλαδή, αν ο αλγόριθμος υπολογίσει ένα ελαφρύτερο (από το ήδη υπολογισμένο) μονοπάτι για τον κόμβο  $y$ , τότε σημειώνει το κόστος του ( $d[y]$ ) και τον αμέσως προηγούμενο κόμβο του νέου υπολογισμένου μονοπατιού ( $prev[y]$ ). Με την αλλαγή του  $d[y]$  αλλάζει και η θέση του κόμβου  $y$  στην ουρά προτεραιότητας  $Q$ . Για την ακρίβεια, μεγαλώνει η προτεραιότητα του  $y$ , αφού κάθε νέα τιμή του  $d[y]$  είναι πάντα μικρότερη από την προηγούμενη. Αφού ο αλγόριθμος εξετάσει όλους τους γείτονες του  $x$  που δεν ανήκουν στο σύνολο  $S$ , εισάγει στο  $S$  τον κόμβο με το ελάχιστο  $d[*]$  από όλους όσους δεν ανήκουν στο  $S$ . Έπειτα, ο αλγόριθμος επιλέγει πάλι τον πρώτο σε προτεραιότητα κόμβο από την ουρά  $Q$  και επαναλαμβάνει τα βήματα αυτά μέχρι να αδειάσει η  $Q$ . Όταν πλέον η  $Q$  θα έχει αδειάσει, ο αλγόριθμος θα έχει βρει τα ελάχιστα μονοπάτια από τον κόμβο  $s$  προς τους όλους τους υπόλοιπους και τα κόστη τους. Ο αλγόριθμος είναι άπληστος (greedy): σε κάθε βήμα, εξετάζει μόνο τους γειτονικούς ενός κόμβου (τοπικότητα). Βρίσκει τον κόμβο για τον οποίο έχει υπολογίσει την ελάχιστη διαδρομή και τον εισάγει στο σύνολο  $S$ , χρησιμοποιώντας πληροφορίες από προηγούμενα βήματα (σύνθεση). Στο τέλος δίνει ένα αποτέλεσμα για όλους τους κόμβους.

Η χρονική πολυπλοκότητα του αλγόριθμου ποικίλλει ανάλογα με τον γράφο και τις δομές δεδομένων που χρησιμοποιούνται για την υλοποίησή του. Στη γενική περίπτωση, ο αλγόριθμος χρειάζεται χρόνο  $O(|V| m + (|V| + |E|) k)$ , όπου  $V$  το σύνολο των κόμβων,  $E$  το σύνολο των ακμών του γράφου,  $m$  ο χρόνος που χρειάζεται για την εξαγωγή του μη-επεξεργασμένου κόμβου με την ελάχιστη ετικέτα απόστασης και  $k$  ο χρόνος που χρειάζεται για την αλλαγή της ετικέτας απόστασης. Στην πιο απλή περίπτωση όπου οι μη-επεξεργασμένοι κόμβοι αποθηκεύονται σε μια λίστα ή σε έναν πίνακα, η αναζήτηση του στοιχείου με το μικρότερο κλειδί γίνεται γραμμικά, εξετάζοντας ένα προς ένα όλα τα στοιχεία. Επομένως, η χρονική πολυπλοκότητα του αλγορίθμου είναι  $O(|V|^2 + |E|) = O(|V|^2)$ .

Ο χρόνος αυτός μπορεί να μειωθεί κι άλλο με τη χρήση μιας ουράς προτεραιότητας για την αποθήκευση των κόμβων που είναι μη-επεξεργασμένοι. Για παράδειγμα, στην περίπτωση μιας υλοποίησης ουράς προτεραιότητας με δυαδικό σωρό, η εξαγωγή ελάχιστου στοιχείου και η αλλαγή του κλειδιού ενός στοιχείου παίρνουν χρόνο  $O(\log n)$ , όπου  $n$  το πλήθος των στοιχείων του σωρού. Έτσι η πολυπλοκότητα του αλγορίθμου μειώνεται σε  $O((|V| + |E|) \log |V|)$ . Για αραιούς γράφους η πολυπλοκότητα ανάγεται σε  $O(|E| \log |V|)$ .

#### 4.3.2 Τροποποιημένος αλγόριθμος Dijkstra

Στην πλοήγηση οχημάτων δεν είναι απαραίτητη η εύρεση της βέλτιστης διαδρομής από τον κόμβο εκκίνησης προς όλους τους κόμβους του οδικού δικτύου. Ο αλγόριθμος Dijkstra μπορεί να τροποποιηθεί έτσι ώστε να φτάνει σε τερματισμό όταν βρεθεί ο κόμβος προορισμού. Για να γίνει αυτό χρησιμοποιείται μια διπλά διασυνδεδεμένη λίστα. Κάθε κόμβος στη λίστα αυτή, έχει έναν δείκτη προς τον πρόγονό του και έναν ή περισσότερους δείκτες προς τους απογόνους του ή το κενό. Επίσης, γίνεται χρήση μιας συνάρτησης αξιολόγησης (συνάρτηση κόστους) για τον υπολογισμό της αξίας κάθε κόμβου που δημιουργείται. Η συνάρτηση αυτή, κατευθύνει τον αλγόριθμο ώστε να αναζητήσει πρώτα τους κόμβους που θεωρούνται πιο κατάλληλοι για να οδηγηθεί σε βέλτιστη λύση. Για έναν κόμβο  $n$  η συνάρτηση αξιολόγησης ορίζεται ως  $g(n)$ , όπου  $g(n)$  είναι το μέτρο του πραγματικού κόστους της μετάβασης από τον τρέχοντα κόμβο προς τον κόμβο  $n$ . Ο καθορισμός της συνάρτησης εξαρτάται από το σχεδιαστή του συστήματος. Κατά την εκτέλεση του αλγορίθμου μετατρέπεται κάθε κόμβος σε μια δομή δεδομένων που δείχνει στον επόμενο κόμβο σε μια από τις δυο συνδεδεμένες λίστες, *Open* και *Closed*:

- *Open*: λίστα κόμβων που έχουν επισκεφθεί αλλά δεν έχουν παραχθεί όλοι οι απόγονοί του
- *Closed*: λίστα κόμβων που δεν έχουν επισκεφθεί ακόμα

Ο αλγόριθμος θα βρει τη βέλτιστη διαδρομή από μια δοσμένη αφετηρία προς έναν δοσμένο προορισμό. Στον αλγόριθμο αυτό ελέγχεται για τον κάθε απόγονο αν βρίσκεται ήδη στη λίστα *Open* ή *Closed*, για να αποφευχθεί η περίπτωση κατά την οποία ένας κόμβος εμφανίζεται παραπάνω από μια φορά. Είναι πιθανό περισσότερες από μια διαδρομές να φτάνουν στον προορισμό διερχόμενες από τον ίδιο κόμβο ή διασταύρωση. Το γεγονός ότι ένας κόμβος μπορεί να έχει διαπεραστεί περισσότερες από μια φορές κατά τη διάρκεια της αναζήτησης, μεταφράζεται στον πραγματικό κόσμο ως η ύπαρξη πολλαπλών διαδρομών που μπορεί να διέρχονται από έναν οδικό κόμβο. Μια από τις διαδρομές αυτές θα έχει το ελάχιστο κόστος.

Αρχικά η λίστα *Open* περιέχει μόνο τον κόμβο αφετηρίας  $s$  με μηδενικό κόστος  $g(s)$ , η λίστα *Closed* είναι άδεια και τα κόστη όλων των υπολοίπων κόμβων είναι ίσα με το άπειρο,  $g[n] = +\infty$ . Αν δεν υπάρχει κανένας κόμβος στη λίστα *Open* ο αλγόριθμος τερματίζει ανεπιτυχώς. Διαφορετικά επιλέγεται ο κόμβος από τη λίστα *Open* με το μικρότερο κόστος  $g(n)$  και ονομάζεται *BEST*. Έπειτα, αφαιρείται από τη λίστα *Open* και τοποθετείται στη λίστα *Closed*. Γίνεται έλεγχος για το αν ο *BEST* είναι ο κόμβος προορισμού. Αν είναι, διασχίζονται όλοι οι κόμβοι της λίστας *Open* προς τα πίσω μέχρι τον κόμβο αφετηρίας και ο αλγόριθμος τερματίζει επιστρέφοντας τη λύση. Αν δεν είναι ο κόμβος προορισμού, παράγονται όλοι οι διάδοχοι του *BEST* και για κάθε διάδοχο κόμβο  $n$  ακολουθούνται τα παρακάτω βήματα:

- Υπολογίζεται το κόστος του κόμβου  $n$ :  $g(n) = g(BEST) + g(BEST, n)$ , όπου  $g(BEST)$  είναι το κόστος του *BEST* και  $g(BEST, n)$  είναι το κόστος για τη μετάβαση από τον *BEST* στον κόμβο  $n$ .
- Αν ο κόμβος  $n$  ταυτίζεται με κάποιον κόμβο, που βρίσκεται ήδη στη λίστα *Open*, ελέγχεται αν ο κόμβος αυτός έχει μικρότερο κόστος  $g(n)$  από τον κόμβο της λίστας με τον οποίον ταυτίζεται. Αν ναι, γίνεται αντικατάσταση του κόστους του κόμβου της λίστας με το κόστος του κόμβου  $n$  και γίνεται ο νέος *BEST*.

- Αν ο κόμβος  $n$  ταυτίζεται με κάποιον κόμβο, που βρίσκεται ήδη στη λίστα *Closed*, ελέγχεται αν ο κόμβος αυτός έχει μικρότερο κόστος  $g(n)$  από τον κόμβο της λίστας με τον οποίον ταυτίζεται. Αν ναι, γίνεται αντικατάσταση του κόστους του κόμβου της λίστας με το κόστος του κόμβου  $n$ , γίνεται ο νέος *BEST* και μεταφέρεται στη λίστα *Open*.
- Αν ο  $n$  δεν υπάρχει σε καμία από τις δύο λίστες τότε γίνεται αυτός *BEST* και τοποθετείται στη λίστα *Open*, και η διαδικασία επαναλαμβάνεται από την αρχή.

Ο αλγόριθμος Dijkstra χρησιμοποιείται για να βρει λύση σε έναν δοσμένο γράφο ή για να υπολογίσει εκ των προτέρων διαδρομές off-line, οι οποίες και αποθηκεύονται στη μνήμη. Απεναντίας, η τροποποιημένη έκδοση χρησιμοποιείται όταν είναι απαραίτητη η πλοήγηση σε πραγματικό χρόνο.

Έστω  $d$  το βάθος της αναζήτησης της συντομότερης λύσης από τον κόμβο εκκίνησης προς τον κόμβο προορισμού. Ως βάθος αναζήτησης ορίζεται ως ο αριθμός των επιπέδων ενός δέντρου, που χρειάζεται η διαδικασία αναζήτησης για να βρει μια λύση. Έτσι η πολυπλοκότητα του τροποποιημένου αλγόριθμου Dijkstra είναι της τάξης  $O(b^d)$ , όπου  $b$  είναι ο παράγοντας διακλάδωσης.

#### 4.3.3 Ο αλγόριθμος Bellman – Ford

Ο αλγόριθμος Bellman-Ford έχει σχεδιαστεί ειδικά για να λύνει το πρόβλημα της εύρεσης συντομότερου μονοπατιού σε ένα γράφο ο οποίος περιέχει και αρνητικά βάρη ακμών. Ο αλγόριθμος διαχειρίζεται την ανικανότητα του αλγόριθμου Dijkstra να δουλέψει με αρνητικές ακμές, με την προσθήκη προστασίας η οποία αποτρέπει το πέρασμα του μονοπατιού μέσα από αρνητικό κύκλο. Αντί να επιλέγεται το ελάχιστο κόστος κόμβου σε κάθε επανάληψη και να ενημερώνονται όλοι οι γειτονικοί κόμβοι, όπως συμβαίνει στον Dijkstra αλγόριθμο, με τον αλγόριθμο Bellman-Ford, ενημερώνονται όλες οι ακμές που υπάρχουν στον γράφο. Η διαδικασία αυτή αποτρέπει την επ' αόριστον επιλογή του ίδιο ελαχίστου κόμβου. Ο χρόνος που απαιτείται για την ενημέρωση του αλγορίθμου είναι ανάλογος με το μέγιστο μήκος του συντομότερου μονοπατιού που μπορεί να επιτευχθεί. Με την ενημέρωση  $n$  φορές, όπου  $n$  είναι ο αριθμός των κόμβων στο γράφο, ο αλγόριθμος Bellman-Ford μπορεί να βρει την ιδανικότερη λύση.

Συγκριτικά με τον αλγόριθμο Dijkstra, ο αλγόριθμος Bellman-Ford, παρουσιάζει χειρότερο χρόνο τρεξίματος, κι αυτό γιατί πρέπει να ενημερώνει κάθε φορά όλες τις πιθανές ακμές στο γράφο, αντί να ενημερώνονται μόνο οι γειτονικές ακμές του κόμβου με το ελάχιστο κόστος. Έτσι, η συνολική πολυπλοκότητα του αλγορίθμου είναι της τάξης του  $O(N^3)$ . Λόγω της υψηλής του πολυπλοκότητας, ο αλγόριθμος Bellman-Ford θα πρέπει να χρησιμοποιείται μόνο σε γραφήματα τα οποία περιέχουν αρνητικά βάρη ακμών.

#### 4.3.4 Ευριστική αναζήτηση & ο αλγόριθμος $A^*$

Η ευριστική αναζήτηση είναι μια πληροφορημένη στρατηγική αναζήτησης, γεγονός που σημαίνει ότι διαθέτει πληροφορίες σχετικά με τον αριθμό των βημάτων ή το κόστος από την αρχική και την παρούσα κατάσταση, μέχρι την τελική. Οι πληροφορίες αυτές χρησιμοποιούνται για τη βελτίωση της αποδοτικότητας της διαδικασίας αναζήτησης, ενώ ενδέχεται να υστερούν σε θέματα πληρότητας. Ο λόγος που πρέπει να γίνει μια αναζήτηση είναι το ότι δεν είναι γνωστή από την αρχή η κατεύθυνση κίνησης. Η ευριστική πληροφορία βοηθάει στον καθορισμό του περισσότερο «υποσχόμενου» κόμβου, ποιοι διάδοχοι θα παραχθούν και ποια άσχετα κλαδιά θα περικοπούν από το δέντρο αναζήτησης, με αποτέλεσμα να γίνει πιο αποδοτική η αναζήτηση. Ο σκοπός της ευριστικής πληροφορίας είναι η παροχή μιας εκτίμησης της απόστασης του κόμβου προορισμού από τον τρέχοντα κόμβο, έτσι ώστε το σύστημα να μπορεί να καθορίσει πόσο πιθανό είναι ο συγκεκριμένος κόμβος να περιλαμβάνεται στη βέλτιστη λύση.

Οι ευριστικές πληροφορίες είναι αποτελεσματικές στο βαθμό που δείχνουν προς τη σωστή κατεύθυνση, αλλά μπορούν περιστασιακά να οδηγήσουν σε αναζήτηση ενός μη αποδεκτού κόμβου, ή σε αδιέξοδο για μια συγκεκριμένη διαδρομή. Παρά το ενδεχόμενο αυτό πρόβλημα, η διαδικασία θα οδηγήσει εν τέλει στον κόμβο προορισμού, εφόσον αυτός υπάρχει. Η χρήση καλά ορισμένων ευριστικών συναρτήσεων προσφέρει τη δυνατότητα εύρεσης καλής, αν όχι βέλτιστης, λύσης σε λιγότερο χρόνο και με μικρότερη κατανάλωση μνήμης σε σχέση με τον απλό ή τον τροποποιημένο Dijkstra.

Ο πιο δημοφιλής αλγόριθμος που εφαρμόζεται πιο συχνά στο σχεδιασμό διαδρομής, είναι ο αλγόριθμος  $A^*$  ο οποίος χρησιμοποιεί τη μέθοδο best-first. Ο αλγόριθμος χρησιμοποιείται και σε πεδία εκτός της πλοήγησης, όπως στη ρομποτική και σε περιοχές ενδιαφέροντος όπου απαιτούνται λύσεις ελαχίστου κόστους. Ο αλγόριθμος  $A^*$  εγγυάται ότι θα βρει τη βέλτιστη διαδρομή, εάν αυτή υπάρχει, με τον όρο να έχει οριστεί ευριστική συνάρτηση που να ικανοποιεί μια συγκεκριμένη συνθήκη.

Αρχικά είναι απαραίτητος ο ορισμός μια καλά ορισμένης ευριστικής συνάρτησης η οποία αξιολογεί κάθε κόμβο αν είναι κατάλληλος ή όχι για επίσκεψη. Η συνάρτηση αυτή θα προτείνει ποια διαδρομή θα πρέπει να ακολουθηθεί πρώτα, όταν περισσότερες από μια διαδρομές είναι διαθέσιμες.

Η ευριστική συνάρτηση αξιολόγησης  $f'(n)$  για τον κόμβο  $n$  ορίζεται ως εξής:

$$f'(n) = g(n) + h'(n)$$

όπου  $g(n)$  είναι το πραγματικό κόστος μετάβασης από τον κόμβο προέλευσης στον τρέχοντα κόμβο  $n$  και  $h'(n)$  είναι μια εκτίμηση του ελαχίστου κόστους μετάβασης από τον τρέχοντα κόμβο  $n$  στον κόμβο προορισμού. Στην περίπτωση που  $h'(n) = 0$  ο αλγόριθμος συμπεριφέρεται όπως ο αλγόριθμος Dijkstra. Ο καθορισμός του κόστους που θα υπολογίζεται από τη συνάρτηση αξιολόγησης εξαρτάται από τον σχεδιαστή του συστήματος.

Αρχικά επιλέγεται ο κόμβος εκκίνησης και ο αλγόριθμος προχωράει σταδιακά, επεκτείνοντας έναν κόμβο σε κάθε βήμα, μέχρι να προκύψει ο κόμβος προορισμού. Σε κάθε βήμα μόνο ο κόμβος με το μικρότερο  $f'(n)$  παράγει απογόνους.

Κατά τη διάρκεια της διαδικασίας αυτής, οι κόμβοι που σχετίζονται με περιορισμούς (π.χ. λάθος κατεύθυνση, αδιέξοδο, κτλ.), εξαιρούνται ώστε να μην συμπεριληφθούν στην τελική διαδρομή. Η ευριστική συνάρτηση εφαρμόζεται για να αξιολογήσει πόσο «ελπιδοφόρος» είναι κάθε κόμβος, αφού πρώτα ελέγξει αν ο κόμβος αυτός έχει παραχθεί ξανά. Η παραπάνω διαδικασία επαναλαμβάνεται, μέχρι να παραχθεί ο κόμβος προορισμού. Εάν η ευριστική συνάρτηση  $h'(n)$  δεν υπερεκτιμά το πραγματικό κόστος για την επίτευξη του κόμβου προορισμού, ο αλγόριθμος  $A^*$  θα καταλήξει στη βέλτιστη λύση, εφόσον αυτή υπάρχει. Αυτό ονομάζεται «ιδιότητα της παραδεκτότητας» (admissibility property).

Η χρονική πολυπλοκότητα του αλγορίθμου εξαρτάται από τη ευριστική συνάρτηση. Στη χειρότερη περίπτωση ενός μη πεπερασμένου χώρου αναζήτησης, το πλήθος των κόμβων που διακλαδώνονται είναι εκθετικό ως προς το μήκος του συντομότερου μονοπατιού  $d$ :

$O(b^d)$ , όπου  $b$  είναι ο παράγοντας διακλάδωσης (μέσος όρος επιτυχών επιλογών ανά κατάσταση). Τα παραπάνω προϋποθέτουν ότι ο κόμβος προορισμού υπάρχει και ότι είναι προσβάσιμος από τον κόμβο εκκίνησης. Σε αντίθετη περίπτωση, και αν ο χώρος των καταστάσεων είναι άπειρος, ο αλγόριθμος δε θα τερματίσει. Η χρονική πολυπλοκότητα όταν ο χώρος αναζήτησης είναι ένα δένδρο, είναι πολυωνυμική και υπάρχει μια μοναδική κατάσταση προορισμού:

$$|h(n) - h^*(n)| = O(\log h^*(n))$$

όπου  $h^*$  είναι η βέλτιστη ευριστική, δηλ. το ακριβές κόστος που απαιτείται για τη μετάβαση από τον κόμβο  $n$  στον κόμβο προορισμού.

#### 4.3.5 Αμφίδρομη αναζήτηση

Η κατανόηση των αλγορίθμων συντομότερης διαδρομής και ευριστικής αναζήτησης οδήγησε στη γενίκευση αυτών και στη δημιουργία αλγορίθμων αμφίδρομης αναζήτησης. Στους αλγορίθμους που παρουσιάστηκαν παραπάνω, η αναζήτηση της συντομότερης διαδρομής πραγματοποιήθηκε από τον κόμβο εκκίνησης προς τον κόμβο προορισμού. Αυτή η αναζήτηση ονομάζεται «αναζήτηση προς τα εμπρός» (forward search). Μια αναζήτηση «προς τα πίσω» (backward search) θα έχει το ίδιο αποτέλεσμα, εφόσον το κόστος κάθε συνδέσμου είναι ίδιο. Αυτό οδήγησε στην ιδέα της αμφίδρομης αναζήτησης.

Η διαδικασία της αμφίδρομης αναζήτησης βασίζεται στον ταυτόχρονο υπολογισμό της καλύτερης διαδρομής από τον κόμβο αφετηρίας προς τον κόμβο προορισμού (forward search) και από τον κόμβο προορισμού προς τον κόμβο αφετηρίας (backward search). Αν ο αλγόριθμος αναζήτησης εκτελείται σε έναν μόνο επεξεργαστή, πρέπει να γίνεται γρήγορη εναλλαγή μεταξύ forward και backward αναζητήσεων. Ο αμφίδρομος αλγόριθμος αναζήτησης ενδέχεται να μειώσει το χώρο αναζήτησης κατά τουλάχιστον 50%, έτσι ώστε ο χρόνος εκτέλεσής του να είναι αρκετά μικρότερος σε σχέση με τη μονόδρομη αναζήτηση.

Αρχικά δημιουργείται η λίστα *forward Open* που περιέχει μόνο τον κόμβο αφετηρίας  $s$  με μηδενικό κόστος  $g(s)$  και μια λίστα *forward Closed* η οποία είναι άδεια και τα κόστη όλων των υπολοίπων κόμβων είναι ίσα με το άπειρο,  $g_{forward}[n] = +\infty$ .

Επίσης, δημιουργείται μια λίστα *backward Open*, που περιέχει μόνο τον κόμβο προορισμού  $f$  με μηδενικό κόστος  $g(f)$  και μια λίστα *backward Closed* που είναι άδεια και τα κόστη όλων των υπολοίπων κόμβων είναι ίσα με το άπειρο,  $g_{backward}[n] = +\infty$ .

Κατά τη διάρκεια εκτέλεσης του αλγορίθμου, γίνεται εναλλαγή μεταξύ της *forward* και *backward* αναζήτησης με βάση κάποιο στατικό ή δυναμικό κριτήριο. Αν δεν υπάρχει κανένας κόμβος στη λίστα *OPEN* της επιλεγμένης κατεύθυνσης αναζήτησης γίνεται εναλλαγή στη λίστα της αντίθετης κατεύθυνσης. Αν δεν υπάρχει κανένας κόμβος και στις δύο λίστες *OPEN*, ο αλγόριθμος τερματίζει ανεπιτυχώς. Διαφορετικά επιλέγεται ο κόμβος με το μικρότερο κόστος  $g(n)$  και τίθεται ως *BEST*. Στη συνέχεια αφαιρείται από τη λίστα *OPEN* και εισάγεται στη λίστα *CLOSED*. Ελέγχεται αν ο *BEST* ικανοποιεί κάποιο κριτήριο τερματισμού. Αν ναι, διασχίζονται όλοι οι κόμβοι της λίστας *OPEN* προς τα πίσω ξεκινώντας από τον *BEST* και καταλήγοντας στον κόμβο αφετηρίας, και προς τα εμπρός μέχρι τον κόμβο προορισμού. Οι δύο λίστες ενώνονται και ο αλγόριθμος τερματίζει επιστρέφοντας την τελική διαδρομή. Αν ο *BEST* δεν ικανοποιεί κάποιο κριτήριο τερματισμού, παράγονται οι διάδοχοί του και για κάθε διάδοχο κόμβο  $n$  ακολουθούνται τα παρακάτω βήματα:

- Υπολογίζεται το κόστος του κόμβου  $n$ :  $g(n) = g(BEST) + g(BEST, n)$ , όπου  $g(BEST)$  είναι το κόστος του *BEST* και  $g(BEST, n)$  είναι το κόστος για τη μετάβαση από τον *BEST* στον κόμβο  $n$ .
- Αν ο κόμβος  $n$  ταυτίζεται με κάποιον κόμβο, που βρίσκεται ήδη στη λίστα *Open*, ελέγχεται αν ο κόμβος αυτός έχει μικρότερο κόστος  $g(n)$  από τον κόμβο της λίστας με τον οποίον ταυτίζεται. Αν ναι, γίνεται αντικατάσταση του κόστους του κόμβου της λίστας με το κόστος του κόμβου  $n$  και γίνεται ο νέος *BEST*.
- Αν ο κόμβος  $n$  ταυτίζεται με κάποιον κόμβο, που βρίσκεται ήδη στη λίστα *Closed*, ελέγχεται αν ο κόμβος αυτός έχει μικρότερο κόστος  $g(n)$  από τον κόμβο της λίστας με τον οποίον ταυτίζεται. Αν ναι, γίνεται αντικατάσταση του κόστους του κόμβου της λίστας με το κόστος του κόμβου  $n$ , γίνεται ο νέος *BEST* και μεταφέρεται στη λίστα *Open*.
- Αν ο  $n$  δεν υπάρχει σε καμία από τις δύο λίστες τότε γίνεται αυτός *BEST* και τοποθετείται στη λίστα *Open*, και η διαδικασία επαναλαμβάνεται από την αρχή.

Υποθέτοντας ότι χρησιμοποιείται ένας μόνο επεξεργαστής, η επιτυχής εφαρμογή του αλγορίθμου αμφίδρομης αναζήτησης απαιτεί δύο επιπλέον συνθήκες: ένα κριτήριο για τον τερματισμό της αναζήτησης και ένα κριτήριο για την εναλλαγή μεταξύ των forward και backward αναζητήσεων. Για παράδειγμα, μπορούμε να καθορίσουμε ένα κριτήριο τερματισμού όταν ένας κόμβος προσεγγισθεί και από τις δυο κατευθύνσεις και ένα στατικό κριτήριο εναλλαγής όταν πραγματοποιηθούν δέκα επαναλήψεις σε κάθε κατεύθυνση. Αν τα παραπάνω κριτήρια έχουν επιλεγθεί κατάλληλα, η αμφίδρομη αναζήτηση θα εξετάσει λιγότερους κόμβους και θα καταλήξει σε μια λύση σε λιγότερο χρόνο συγκριτικά με τη μονόδρομη αναζήτηση.

Όπως είναι λογικό η χρονική πολυπλοκότητα του αλγορίθμου μειώνεται στο μισό της πολυπλοκότητας του αλγορίθμου  $A^*$ , δηλ. είναι της τάξης  $O(b^{d/2})$ . Στην ιδανική περίπτωση οι αναζητήσεις προς τις δυο κατευθύνσεις πρέπει να συναντηθούν στη μέση. Αν υπάρχουν πολλές διαφορετικές διαδρομές από τον κόμβο αφετηρίας στον κόμβο προορισμού, οι αναζητήσεις είναι πιθανόν να μη συναντηθούν στη μέση.

Στη χειρότερη περίπτωση (π.χ. όταν το κριτήριο τερματισμού της ευριστικής αξιολόγησης λαμβάνει υπόψη μόνο το εκτιμώμενο κόστος διαδρομής) η αμφίδρομη ευριστική αναζήτηση αποκτά διπλάσια πολυπλοκότητα σε σχέση με τη μονόδρομη ευριστική αναζήτηση. Επομένως, είναι ιδιαίτερα σημαντικό να γίνεται επιλογή της σωστής ευριστικής συνάρτησης αξιολόγησης και των κριτηρίων τερματισμού της διαδικασίας.

Μια αμφίδρομη ευριστική αναζήτηση πρέπει να ελέγξει σχεδόν τους διπλάσιους κόμβους σε σχέση με μια μονόδρομη ευριστική αναζήτηση για να προσδιορίσει τη βέλτιστη διαδρομή. Αυτό συμβαίνει διότι η αμφίδρομη αναζήτηση πρέπει να συγκρίνει όλες τις ενωμένες διαδρομές κατά την αναζήτηση για να εξασφαλίσει πως η διαδρομή είναι βέλτιστη.

Η αμφίδρομη αναζήτηση είναι καλή επιλογή για την εύρεση συντομότερου μονοπατιού μόνον όταν δεν είναι απαραίτητη η βέλτιστη λύση. Όταν χρησιμοποιείται ο αμφίδρομος αλγόριθμος οι περιορισμοί που θέτονται κατά την βελτιστοποίηση πρέπει να «χαλαρώνουν». Στην περίπτωση αυτή θα βρεθεί μια σχεδόν βέλτιστη λύση πριν βρεθεί η συνολικά βέλτιστη λύση.

#### 4.3.6 *Ιεραρχική αναζήτηση*

Υπάρχει μια φυσική ιεραρχία στους δρόμους του πραγματικού κόσμου, με τους αυτοκινητόδρομους να βρίσκονται στην κορυφή. Αυτό οδήγησε στην ιδέα για την ανάπτυξη ενός πιο αποτελεσματικού αλγορίθμου ιεραρχικής αναζήτησης για ένα δοσμένο δίκτυο. Η βασική ιδέα βρίσκεται στην αναζήτηση σε έναν, αρχικά, «αφαιρετικό» χώρο και όχι σε ολόκληρο το χώρο του προβλήματος. Η επιλογή μιας απλοποιημένης αναπαράστασης αυτού του τύπου οδηγεί σε σημαντική αύξηση της αποτελεσματικότητας του αλγορίθμου.

Ο βασικός λόγος που η αφαίρεση μειώνει την πολυπλοκότητα των προβλημάτων, είναι ότι η συνολική πολυπλοκότητα ισούται με το άθροισμα των πολυπλοκότητων των ανεξάρτητων αναζητήσεων και όχι με το γινόμενο αυτών. Για έναν αλγόριθμο με πολυπλοκότητα τάξης  $O(b^d)$ , με τη χρήση ενός επιπέδου αφαίρεσης η ιεραρχική αναζήτηση μπορεί να μειώσει την πολυπλοκότητα σε  $O(b^{d/2})$ . Στην περίπτωση χρήσης πολλαπλών επιπέδων αφαίρεσης, η βέλτιστη ιεραρχία, για έναν χώρο με  $b^d$  καταστάσεις, θα έχει  $\ln b^d$  επίπεδα αφαίρεσης. Αυτός ο τύπος ιεράρχησης μπορεί να μειώσει την πολυπλοκότητα του αλγορίθμου αναζήτησης από  $O(b^d)$  σε  $O(d \log b)$ . Με άλλα λόγια η βέλτιστη ιεράρχηση έχει λογαριθμικό αριθμό επιπέδων με σταθερό λόγο και μειώνει την πολυπλοκότητα, από εκθετική σε γραμμική.

Για να εφαρμοστεί η ιεραρχική αναζήτηση και να απλοποιηθεί η διαδικασία σχεδιασμού της διαδικασίας εύρεσης συντομότερου μονοπατιού, πρέπει η βάση δεδομένων του χάρτη να κατασκευαστεί ιεραρχικά. Όσο ψηλότερα βρίσκεται το επίπεδο αφαίρεσης τόσο λιγότερες λεπτομέρειες περιέχει, δηλαδή, λιγότερους κόμβους και ακμές.

Ωστόσο, η αναζήτηση σε ένα τέτοιο επίπεδο μπορεί να καταλήξει σε μη ολοκληρωμένη λύση, κάτι που σημαίνει ότι πρέπει να γίνεται εναλλαγή μεταξύ των διαφορετικών επιπέδων κατά τη διαδικασία της αναζήτησης.

Στην πράξη είναι ιδιαίτερα δύσκολο να κατασκευαστεί μια βάση δεδομένων ή να σχεδιαστεί ένας αλγόριθμος ο οποίος να ικανοποιεί τις συνθήκες της βέλτιστης ιεραρχικής αφαίρεσης. Στην πλοήγηση οχημάτων, χρησιμοποιούνται κλάσεις δρόμων για τον καθορισμό μιας ιεραρχίας στον σχεδιασμό της διαδρομής.

## **4.4 Συμπεράσματα**

Οι τεχνικές που έχουν αναπτυχθεί τα τελευταία χρόνια για την επιτάχυνση της διαδικασίας δρομολόγησης σε στατικά οδικά δίκτυα, έχουν εξελιχθεί πολύ τα τελευταία χρόνια. Μια ιδιαίτερη πρόκληση είναι να μειωθεί το χάσμα μεταξύ του θεωρητικού υπόβαθρου και της πράξης. Για παράδειγμα, όταν επιθυμείται να δοθεί ένας ουσιώδης χαρακτηρισμός για δίκτυα που συμπεριφέρονται σωστά και επιτρέπουν αποδεδειγμένα την ύπαρξη του καλύτερου και του χειρότερου σεναρίου εκτέλεσης. Πιο συγκεκριμένα θα ήταν καλό να γνωρίζει κανείς για ποια δίκτυα μπορούν να εφαρμοστούν οι ήδη υπάρχουσες τεχνικές, πχ. για δίκτυα τηλεπικοινωνιών, για το σχεδιασμό VLSI, κοινωνικά δίκτυα, παιχνίδια για ηλεκτρονικού υπολογιστές, γραφικά που προέρχονται από προβλήματα γεωμετρικής δρομολόγησης, κτλ.

Η κύρια ακαδημαϊκή πρόκληση είναι να γίνει μετάβαση σε ένα είδος δρομολόγησης που δεν είναι αποκλειστικά προσκολλημένο στον τομέα της «point-to-point» δρομολόγησης. Οι δημόσιες μεταφορές και τα οδικά δίκτυα που εξαρτώνται από τη διάρκεια διαδρομής είναι μια προφανής γενίκευση που θα πρέπει να συνδυαστεί με την ανανέωση στα βάρη των ακμών μεταξύ των κόμβων. Ο συνδυασμός αυτός θα εξαρτάται από την κυκλοφοριακή συμφόρηση που επικρατεί σε κάθε ακμή και η ανανέωση αυτή θα πρέπει να γίνεται σε πραγματικό χρόνο. Για να γίνει αυτό είναι αναγκαία η ύπαρξη πολλών κριτηρίων βελτιστοποίησης τα οποία πρέπει να αλληλεπιδρούν δυναμικά μεταξύ τους, ώστε να είναι δυνατή η εύρεση της βέλτιστης διαδρομής κάθε φορά και αυτό θα πρέπει να γίνεται όσο το δυνατόν γρηγορότερα. Επιπροσθέτως θα πρέπει να συνυπολογίζονται πολλοί παράγοντες που επηρεάζουν κάθε φορά τη διαδικασία εύρεσης διαδρομής, όπως ζητήματα υλοποίησης σε κινητές συσκευές, διαδικασίες που εξαρτώνται από τον χρήστη και ζητήματα συμβατότητας με τις ήδη υπάρχουσες εφαρμογές. Η δυσκολία στο σημείο αυτό έγκειται στο να βρεθεί μια προσέγγιση για την ομαλή συνεργασία ανάμεσα στο ακαδημαϊκό και στο βιομηχανικό περιβάλλον. [12]

Στο Κεφάλαιο 5 θα παρουσιαστούν κάποια συστήματα πλοήγησης που υπάρχουν και θα περιγραφούν οι μέθοδοι εύρεσης τοποθεσίας που χρησιμοποιούν τα συστήματα αυτά.

### **5.1 Γενικά**

Για να περιηγηθεί κάποιος μέσα σε έναν χώρο πρέπει να είναι ορισμένη η τοποθεσία του. Δύο πολύ σημαντικοί παράγοντες που λαμβάνονται υπ' όψιν κατά τη δημιουργία συστημάτων περιήγησης είναι η ακρίβεια και η χρονική απόκριση. Οι δύο αυτοί παράγοντες συνυπολογίζονται τόσο για σταθερές όσο και για κινητές συσκευές πλοήγησης.

Υπάρχουν δύο διαφορετικοί μέθοδοι για την υλοποίηση συστημάτων πλοήγησης: η «αυτοτοποθέτηση» (self-positioning) και η «τοποθέτηση εξ αποστάσεως» (remote positioning). Στην αυτοτοποθέτηση, η φυσική τοποθεσία ορίζεται από την ίδια τη συσκευή του χρήστη κάνοντας χρήση σημάτων που λαμβάνονται από δορυφόρο. Η τοποθεσία είναι γνωστή στον χρήστη και χρησιμοποιείται από εφαρμογές και υπηρεσίες που ενεργούν στην κινητή συσκευή. Στην τοποθέτηση εξ αποστάσεως, η τοποθεσία ορίζεται σε κάποιο server χρησιμοποιώντας σήματα που εκπέμπονται από τη συσκευή. Η τοποθεσία τότε χρησιμοποιείται από τον server σε ένα λογισμικό σύστημα εντοπισμού, ή μεταδίδεται πίσω στη συσκευή μέσω κάποιας μεθόδου μεταφοράς δεδομένων.

Η επίδοση των συστημάτων τοποθέτησης και πλοήγησης εξαρτάται από τέσσερις παράγοντες: ακρίβεια (accuracy), ακεραιότητα (integrity), διαθεσιμότητα (availability) και συνέχεια (continuity). Η ακρίβεια ενός συστήματος μετράει την πιθανότητα να προκύψει κάποιο σφάλμα κατά την εύρεση τοποθεσίας σε οποιαδήποτε χρονική στιγμή. Η ακεραιότητα του συστήματος μετράει την πιθανότητα το σφάλμα που εμφανίστηκε να βρίσκεται εντός των επιτρεπτών ορίων παρουσίας σφαλμάτων που έχουν οριστεί. Η διαθεσιμότητα είναι ένα μέτρο για την ικανότητα του συστήματος να συνδυάζει ταυτόχρονα την ακρίβεια και την ακεραιότητά του. Η συνέχεια του συστήματος είναι η ικανότητά του να μετράει το ελάχιστο χρονικό διάστημα για το οποίο η υπηρεσία είναι διαθέσιμη στον χρήστη. [8]



## 5.2 Μέθοδοι εύρεσης τοποθεσίας

### 5.2.1 Καθολικό δορυφορικό σύστημα πλοήγησης - *Global Navigation Satellite*

#### *System (GNSS)*

Το σύστημα GNSS, που συχνά αναφέρεται στο αμερικάνικο σύστημα NAVSTAR και είναι αναγνωρίσιμο ως GPS (Global Positioning System) είναι το πιο ακριβές σύστημα για την αναγνώριση τοποθεσίας χρήστη. Το GPS χρησιμοποιείται για την παρακολούθηση, τοποθέτηση, εντοπισμό και πλοήγηση οχημάτων εδώ και πολλά χρόνια. Η βασική αρχή είναι η μέτρηση της απόστασης από τον δέκτη του χρήστη μέχρι έναν από τους τουλάχιστον 4 δορυφόρους, για τους οποίους είναι γνωστή η τοποθεσία τους στην τροχιά γύρω από τη γη. Αυτό επιτρέπει διδιάστατη και τρισδιάστατη τοποθέτηση παγκοσμίως ανεξάρτητα από τις καιρικές συνθήκες και τη χρονική στιγμή. Η «ψευδοαπόσταση» (pseudorange) ορίζει την γεωμετρική απόσταση μεταξύ του αναμεταδότη και του δέκτη, που έχει ανακτηθεί από τη χρονική διαφορά ανάμεσα στη στιγμή που το σήμα στέλνεται στον δορυφόρο, μέχρι τη στιγμή που επιστρέφεται η απάντηση στον δέκτη. Ως «φάση του φέροντος» (carrier phase) ορίζεται η διαφορά ανάμεσα στη φάση του εισερχόμενου φέροντος σήματος και τη φάση του σήματος αναφοράς που παράγεται στον δέκτη.

Υπάρχουν δύο βασικές λειτουργίες τοποθέτησης στα συστήματα GPS: η απόλυτη τοποθέτηση και η σχετική τοποθέτηση. Στην απόλυτη τοποθέτηση ένας δέκτης μπορεί να παρακολουθεί τις ψευδοαποστάσεις από πολλαπλούς δορυφόρους για να καθορίσει την τοποθεσία του χρήστη. Στην σχετική τοποθέτηση γίνεται χρήση των διαφορικών τεχνικών GPS (differential GPS: DGPS). Η τεχνολογία DGPS περιλαμβάνει τουλάχιστον δύο δέκτες, από τους οποίους ο ένας είναι ο δέκτης αναφοράς και οι συντεταγμένες του πρέπει να είναι γνωστές. Ο δεύτερος δέκτης ονομάζεται δέκτης του χρήστη και οι συντεταγμένες του καθορίζονται με βάση τον δέκτη αναφοράς, παραγοντοποιώντας τις διαφορές μετρήσεις στους ίδιους δορυφόρους που παρατηρούνται από τους δέκτες αναφοράς και χρήστη.

Με τη χρήση της ψευδοαπόστασης επιτυγχάνεται ακρίβεια της τάξης 1-5 μέτρα για τη λειτουργία της απόλυτης τοποθέτησης, ενώ για τη λειτουργία της σχετικής τοποθέτησης η ακρίβεια μπορεί να φτάσει και τα 50 εκατοστά. Αν συμπεριληφθούν και οι μετρήσεις από τις φάσεις του φέροντος η ακρίβεια μπορεί να αγγίξει και τα όρια του χιλιοστού.

Υπάρχουν πολλές συσκευές που υποστηρίζουν GPS, οι τιμές των οποίων κυμαίνονται από 100\$ έως 10.000\$ και χρησιμοποιούνται για διάφορους σκοπούς. Υπάρχουν όμως μόνο δύο παράγοντες που επηρεάζουν την ευαισθησία τους:

- Τα συστήματα GPS χρειάζονται σήματα έντασης τουλάχιστον -130dBm (decibel - milliwatts), γεγονός που ενδεχομένως να προκαλέσει πρόβλημα σε δασικές περιοχές ή σε περιοχές με πυκνή βλάστηση.
- Μέσα σε κτήρια, τούνελ ή συμπαγείς εγκαταστάσεις γενικότερα, το σήμα GPS απορροφάται σχεδόν εξ ολοκλήρου. Για να χρησιμοποιηθεί ένα σύστημα που βασίζεται σε δορυφόρο, σε εσωτερικούς χώρους θα πρέπει να βελτιωθεί η διαδικασία λήψης του σήματος, είτε βελτιώνοντας τον ίδιο τον δέκτη, είτε με την εγκατάσταση τοπικών GPS αναμεταδοτών.

### **5.2.2 Υποβοηθούμενο καθολικό σύστημα εύρεσης τοποθεσίας – Assisted GPS**

Τα Assisted-GPS (A-GPS) συστήματα παρέχουν στον GPS δέκτη δεδομένα ή πληροφορία που σε διαφορετική περίπτωση θα έπρεπε να τα «κατεβάσει» από τον δορυφόρο. Η παροχή αυτή βελτιώνει σημαντικά το χρόνο που απαιτείται για την λήψη των δεδομένων αυτών στο δέκτη, κάνοντας τους ίδιους τους δέκτες γρηγορότερους. Οποιοδήποτε σύστημα GPS μπορεί με τις κατάλληλες τροποποιήσεις να επωφεληθεί από τα δεδομένα του Assisted-GPS. Με ένα A-GPS σύστημα, ο δέκτης έχει τη δυνατότητα να ενοποιήσει τα σήματα που λαμβάνει από τους δορυφόρους 10 φορές γρηγορότερα, πράγμα που οδηγεί σε κέρδος της τάξης των 10dB.

### **5.2.3 Κυψελοειδή δίκτυα – Cellular networks**

Τα συστήματα επικοινωνίας κινητών τηλεφώνων όπως το GSM (Global Standard for Mobile Communication) ή το UMTS (Universal Mobile Telecommunication System), βασίζονται σε ένα σύνολο από κυψελοειδή δίκτυα. Εδώ και μερικά χρόνια οι υπηρεσίες που βασίζονται στην τοποθεσία του χρήστη, (location-based services, LBS) παρέχονται μέσω αυτών των κυψελοειδών δομών. Η διαδικασία αυτή επιτυγχάνεται με τη χρήση μιας ικανοποιητικής σχέσης μεταξύ του παρόχου της κυψελοειδούς υπηρεσίας, των κυψελοειδών δικτύων και των τερματικών των χρηστών, τα οποία λειτουργούν σε συγχρονισμό για να εντοπίσουν την τοποθεσία του χρήστη. Το ζητούμενο είναι να εντοπιστεί η τοποθεσία του χρήστη με τη μεγαλύτερη ακρίβεια και με τον μικρότερο χρόνο απόκρισης. Ο πιο δημοφιλής τρόπος είναι η τεχνική εύρεσης τοποθεσίας με τη χρήση ραδιοσημάτων, η οποία χρησιμοποιεί παραμέτρους των ραδιοκυμάτων. Τα ραδιοκύματα μεταφέρονται από τη συσκευή του χρήστη προς τους σταθμούς αναφοράς για να παράξουν την τοποθεσία του. Οι παράμετροι του σήματος που χρησιμοποιούνται είναι: η γωνία άφιξης (angle of arrival - AOA), ο χρόνος άφιξης (time of arrival - TOA), η ισχύς του σήματος και η χρονική διαφορά άφιξης των σημάτων (time difference of arrival - TDOA). [9]

#### **5.2.3.1 Γωνία άφιξης (Angle of Arrival - AOA)**

Η μέθοδος AOA κάνει χρήση πολλών σταθμών βάσης για να προσεγγίσει την τοποθεσία του χρήστη. Σε ένα σύστημα AOA, απαιτούνται δύο σταθμοί βάσης, των οποίων οι συντεταγμένες είναι γνωστές, για να προσδιοριστεί η γωνία από την οποία λήφθηκε το σήμα του χρήστη. Η γωνία προσδιορίζεται με τη χρήση μιας κατευθυνόμενης ακτίνας που εκπέμπεται από την κεραία, μέχρις ότου το σήμα αποκτήσει την μεγαλύτερή του ισχύ. Η τοποθεσία καθορίζεται με την παρεμβολή των τοποθεσιών για κάθε έναν από τους AOA σταθμούς βάσης, η οποία θα πρέπει να είναι μια ευθεία γραμμή. Αν ο χρήστης και οι σταθμοί βάσης δεν βρίσκονται στο ίδιο επίπεδο, τότε είναι απαραίτητη η χρήση τριών κεραίων. Η χρήση περισσότερων σταθμών βάσης μπορεί να βελτιώσει κατά πολύ την ακρίβεια των υπολογισμών τοποθεσίας. Η συνολική ακρίβεια του συστήματος εξαρτάται από τη διάδοση του σήματος, την ακρίβεια των κατευθυνόμενων κεραίων και την απόσταση μεταξύ της συσκευής και των κεραίων.

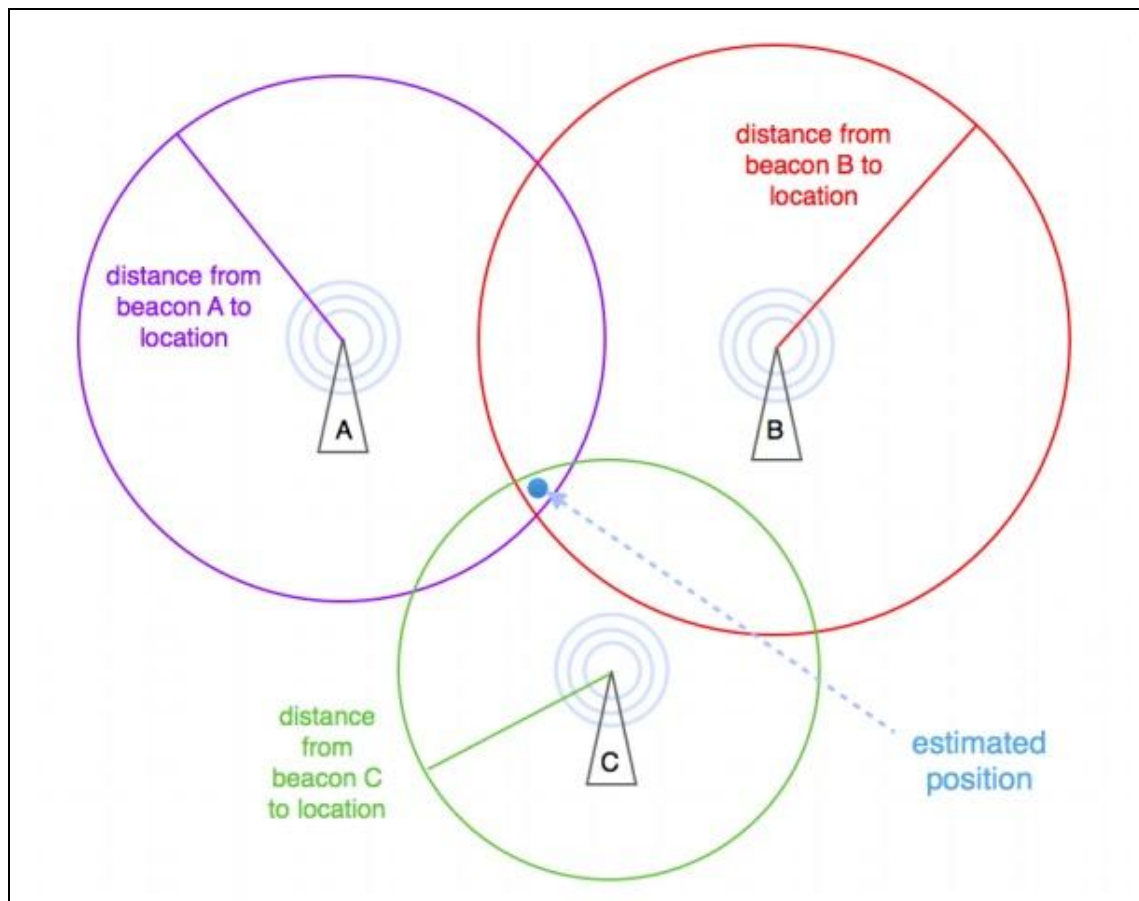
#### **5.2.3.2 Γωνιακή διαφορά άφιξης (Angle Difference of Arrival)**

Η μέθοδος ADOA είναι μια αυτοπροσδιοριζόμενη μέθοδος η οποία χρησιμοποιεί πολλαπλούς σταθμούς βάσης για να προσεγγίσει την τοποθεσία του χρήστη. Σε ένα ADOA σύστημα εύρεσης τοποθεσίας η συσκευή που είναι εφοδιασμένη με προσανατολιζόμενη κεραία πρέπει να καθορίσει τη σχετική γωνία από την οποία έχουν προέλθει τα σήματα από τους τρεις σταθμούς βάσης, των οποίων η τοποθεσία είναι γνωστή.

Η προσθήκη ενός επιπλέον σταθμού βάσης είναι απαραίτητη λόγω της έλλειψης προσανατολισμού του χρήστη. Αν ο χρήστης και οι σταθμοί βάσης δε βρίσκονται στο ίδιο επίπεδο, τότε απαιτείται η χρήση τρισδιάστατων κατευθυνόμενων κεραιών. Σε διαφορετική περίπτωση οι ακτίνες δύο διαστάσεων επαρκούν. Η χρήση περισσότερων σταθμών βάσης αυξάνει ραγδαία την αποδοτικότητα. Η συνολική ακρίβεια του συστήματος εξαρτάται από τη διάδοση του σήματος και την ακρίβεια των κατευθυνόμενων κεραιών.

### 5.2.3.3 Χρόνος άφιξης (Time of Arrival - TOA)

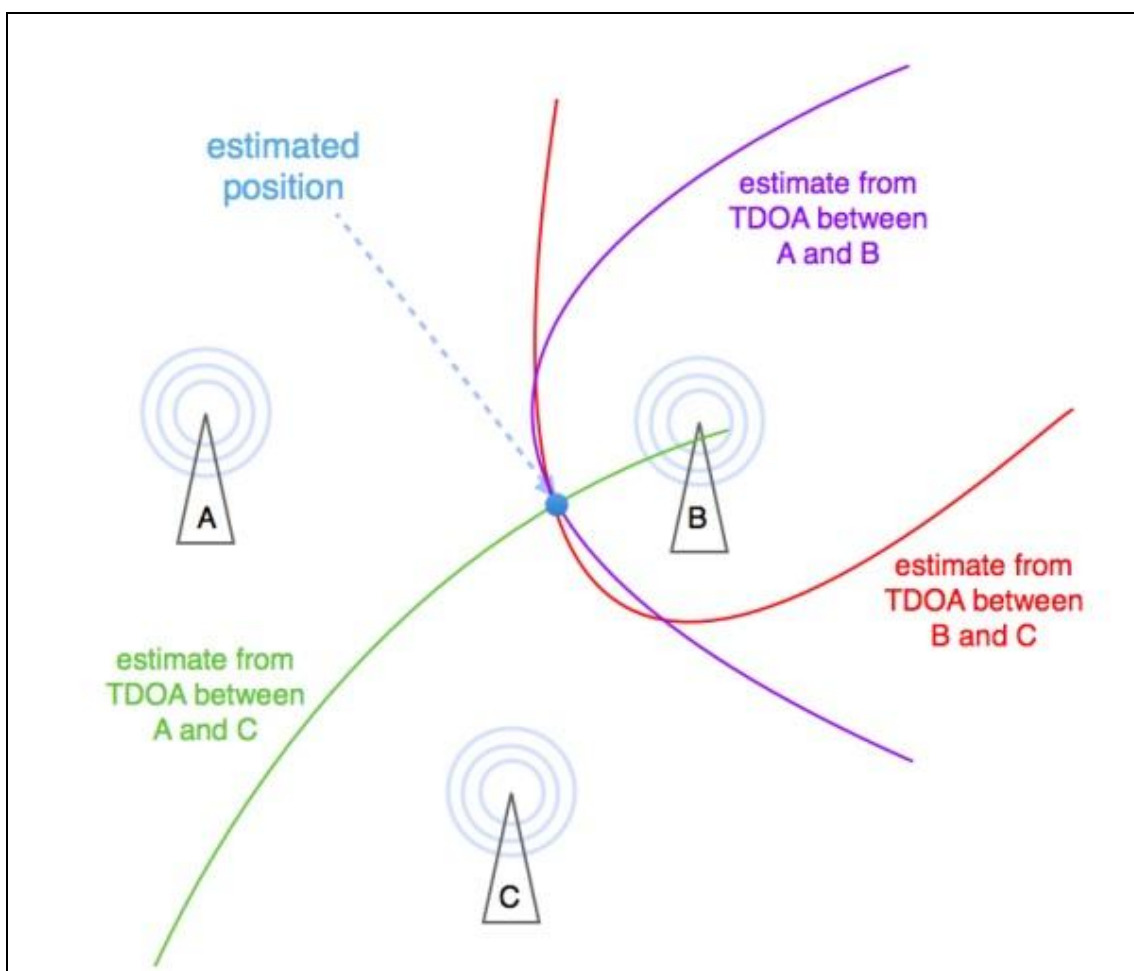
Η TOA είναι μια μέθοδος εύρεσης τοποθεσίας που χρησιμοποιεί μια μορφή τριγωνισμού (triangulation) για να καθορίσει την τοποθεσία του χρήστη. Η απόσταση υπολογίζεται από την απόλυτη τιμή του χρόνου που απαιτείται για να φτάσει ένα σήμα από τον αναμεταδότη στον δέκτη. Για να είναι εφικτός ο τριγωνισμός, θα πρέπει να είναι γνωστή η τοποθεσία καθενός από τους τρεις σταθμούς βάσης. Σε σύγχρονα συστήματα, ο χρόνος μετάδοσης μπορεί να μετατραπεί απ' ευθείας σε απόσταση, αλλά αυτό απαιτεί ο δέκτης να γνωρίζει την ακριβή χρονική στιγμή που έγινε η μετάδοση. Στα ασύγχρονα συστήματα, είναι απαραίτητη η χρήση πρωτοκόλλων εκπομπής και λήψης, που μετατρέπουν το χρόνο σε απόσταση. Με γνωστές τις τρεις αποστάσεις, μπορεί να γίνει χρήση του τριγωνισμού για τον υπολογισμό της τοποθεσίας του χρήστη. Η ακρίβεια της μεθόδου εξαρτάται από τα ενδεχόμενα σφάλματα κατά τη διαδικασία της μετάδοσης και από την ακρίβεια των μετρήσεων των χρόνων.



Εικόνα 5.1 - Μέθοδος TOA

#### 5.2.3.4 Χρονική διαφορά άφιξης (Time Difference of Arrival)

Η μέθοδος TDOA είναι παρόμοια με την TOA με τη μόνη διαφορά ότι η TDOA απαιτεί συγχρονισμό μεταξύ των σταθμών βάσης, αλλά δεν απαιτεί συγχρονισμό μεταξύ των σταθμών βάσης και του χρήστη. Επιπλέον, ο χρήστης δε χρειάζεται να κάνει μετάδοση σήματος στους σταθμούς βάσης. Η τοποθεσία καθορίζεται με την παρεμβολή των τοποθεσιών των χρονικών διαφορών άφιξης, η οποία έχει τη μορφή υπερβολής στο χώρο δύο διαστάσεων και κυρτής ελλειπτικής επιφάνειας στον τρισδιάστατο χώρο. Στα συστήματα TDOA, ο αριθμός των σταθμών βάσης πρέπει να είναι κατά ένα περισσότερα από τον αριθμό των διαστάσεων. Για την ακρίβεια της μεθόδου αυτής ισχύουν τα ίδια κριτήρια με την TOA μέθοδο.



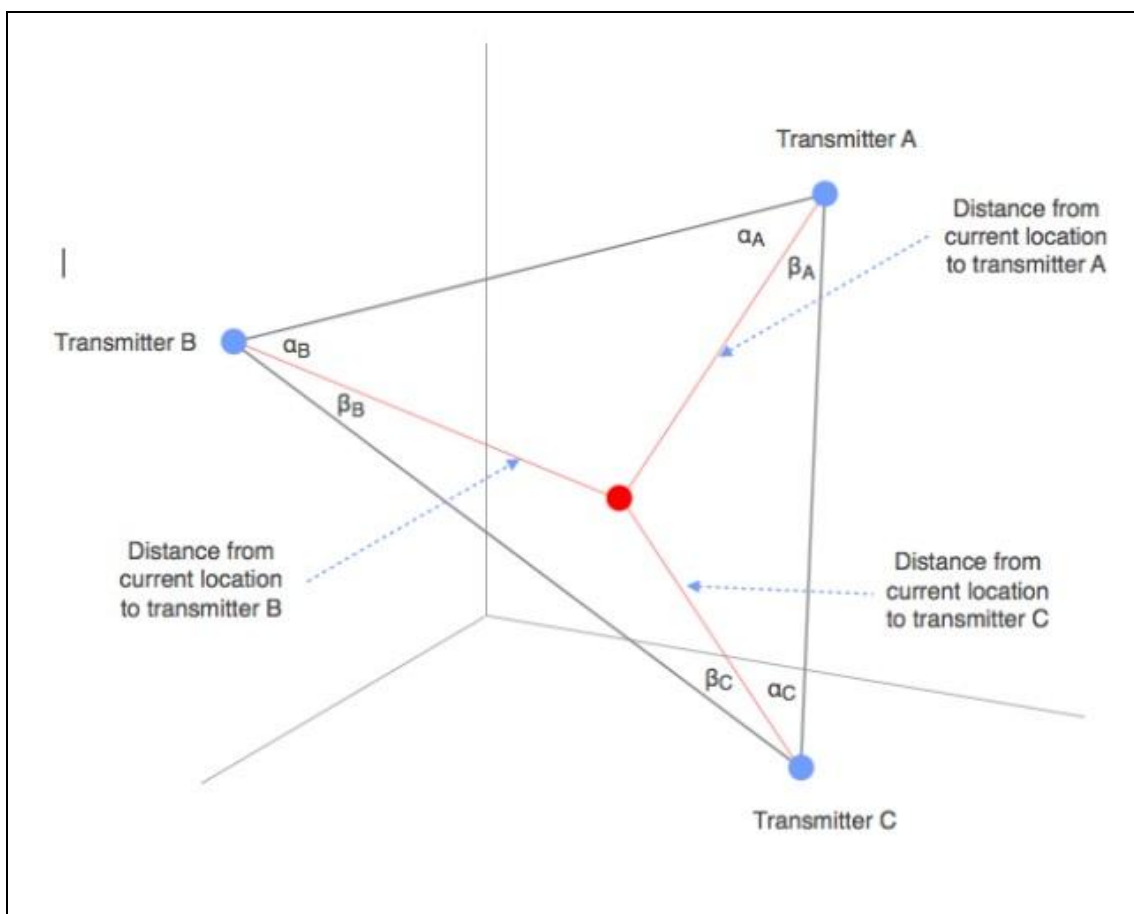
Εικόνα 5.2 - Μέθοδος TDOA

### 5.2.3.5 Η μέθοδος της «κυψέλης προέλευσης» - Cell-of-origin

Με τη μέθοδο cell-of-origin, η κινητή συσκευή αποφασίζει για το ποια κεραία που είναι πιο κοντά σε αυτήν, μεταφέρει το σήμα στο χρήστη. Η κάλυψη μιας κεραίας διαφέρει από περιοχή σε περιοχή, δηλαδή σε αγροτικές περιοχές μπορεί να παρέχει κάλυψη σε ακτίνα 35 km, ενώ σε αστικές περιοχές η κάλυψη μειώνεται σε μερικές εκατοντάδες μέτρα. Δεδομένου ότι ένα GSM σύστημα επιλέγει αυτόματα την καλύτερη διαθέσιμη κυψέλη, μπορεί να χρησιμοποιηθεί η πληροφορία της κυψέλης αυτής (με την οποία είναι συνδεδεμένη η κινητή συσκευή) για να βρεθεί η τοποθεσία του χρήστη.

### 5.2.3.6 Τριγωνισμός - Triangulation

Σε ένα περιβάλλον με γνωστό τον παράγοντα της απώλειας κατά τη διάδοση του σήματος, μπορεί εύκολα από την ισχύς του σήματος να προκύψει η απόστασή μεταξύ πομπού και δέκτη. Σε πραγματικά δεδομένα, για να είναι ακριβής η μετατροπή αυτή, θα πρέπει να λαμβάνονται υπ' όψιν παράγοντες όπως η ισχύς της κεραίας, το κέρδος της και αντικείμενα που ενδεχομένως να παρεμβάλλονται στην πορεία του σήματος. Η ακρίβεια του τριγωνισμού εξαρτάται από την ακρίβεια με την οποία γίνεται η εκτίμηση των απωλειών κατά τη μετάδοση του σήματος. Επίσης, είναι εύκολο να υλοποιηθεί.



Εικόνα 5.3 - Μέθοδος του τριγωνισμού

### 5.2.3.7 Μέθοδος εύρεσης τοποθεσίας με τη χρήση αποτυπωμάτων –

#### *Location Fingerprinting*

Με την τεχνική αυτή γίνεται σύγκριση των RSSI (Received Signal Strength Indication) δεδομένων που έχουν ήδη μετρηθεί, με τις αποθηκευμένες αναμενόμενες τιμές από κάποια βάση δεδομένων. Συνήθως, η μέθοδος αυτή χρησιμοποιείται για την εύρεση της τοποθεσίας του χρήστη μέσα σε κτήρια και οι μετρήσεις γίνονται σε ένα αυθαίρετο πρότυπο πλέγμα που περιβάλλει το κτήριο. Για την αναζήτηση στη βάση δεδομένων μπορεί να χρησιμοποιηθεί ένας αλγόριθμος αυτοσυσχέτισης πολλαπλών πινάκων και συνεπώς να γίνει η εκτίμηση της τοποθεσίας. Η μέθοδος αυτή είναι πολύ ακριβής, ωστόσο απαιτεί πολύ χρόνο στην υλοποίησή της.

### 5.2.3.8 Σύνοψη

Στον παρακάτω πίνακα (Πίνακας 5.4) εμφανίζονται συγκεντρωτικά τα πλεονεκτήματα και τα μειονεκτήματα της κάθε μεθόδου που περιγράφηκε προηγουμένως.

Μέθοδος	Πλεονεκτήματα	Μειονεκτήματα
Γωνία άφιξης (AOA)	<ul style="list-style-type: none"><li>Μέτρια έως καλή ακρίβεια με το κατάλληλο hardware</li></ul>	<ul style="list-style-type: none"><li>Απαιτεί κατευθυνόμενες κεραίες</li><li>Απαιτεί γνώση προσανατολισμού</li></ul>
Γωνιακή διαφορά άφιξης (ADOA)	<ul style="list-style-type: none"><li>Δεν απαιτεί γνώση προσανατολισμού</li></ul>	<ul style="list-style-type: none"><li>Είναι αναγκαία η προσθήκη ενός επιπλέον σταθμού βάσης</li></ul>
Χρόνος άφιξης (TOA)	<ul style="list-style-type: none"><li>Αρκετά καλές επιδόσεις σε εσωτερικούς χώρους</li></ul>	<ul style="list-style-type: none"><li>Οι σταθμοί βάσης πρέπει να είναι συγχρονισμένοι</li><li>Συνολικά χαμηλή ακρίβεια</li></ul>
Χρονική διαφορά άφιξης (TDOA)	<ul style="list-style-type: none"><li>Αρκετά καλές επιδόσεις σε εσωτερικούς χώρους</li></ul>	<ul style="list-style-type: none"><li>Συνολικά χαμηλή ακρίβεια</li></ul>
Κυψέλη προέλευσης (Cell-of-Origin)	<ul style="list-style-type: none"><li>Οι σταθμοί βάσης υπάρχουν ήδη (cell towers)</li><li>Η τοποθεσία των σταθμών βάσης δεν μεταβάλλεται</li></ul>	<ul style="list-style-type: none"><li>Δεν παρουσιάζει μεγάλη ακρίβεια</li></ul>
Τριγωνισμός (Triangulation)	<ul style="list-style-type: none"><li>Πολύ απλή μέθοδος</li></ul>	<ul style="list-style-type: none"><li>Απαιτεί καθορισμό των γωνιών</li></ul>
Χρήση αποτυπωμάτων (Location Fingerprinting)	<ul style="list-style-type: none"><li>Μεγάλη ακρίβεια</li></ul>	<ul style="list-style-type: none"><li>Μεγάλος χρόνος υλοποίησης</li></ul>

Πίνακας 5.4 - Πλεονεκτήματα & μειονεκτήματα των μεθόδων εύρεσης τοποθεσίας

#### 5.2.4 Ασύρματο τοπικό δίκτυο - WLAN (Wireless Local Area Network)

Λόγω της μεγάλης τους ανάπτυξης, τα ασύρματα δίκτυα έχουν γίνει ο ιδανικός τρόπος για την παροχή υπηρεσιών web σε εφαρμογές κινητών για χώρους περιορισμένης κλίμακας, όπως αίθουσες, χώρους πανεπιστημίων, εμπορικά κέντρα, εγκαταστάσεις εργοστασίων και άλλους εσωτερικούς χώρους. Για τα είδη των εφαρμογών αυτών, η εύρεση τοποθεσίας μέσω WLAN είναι μια ενδιαφέρουσα προσέγγιση.

Η προφανής μέθοδος είναι να επιτευχθεί η εύρεση θέσης με τον καθορισμό της απόστασης, με την μέτρηση της καθυστέρησης στη διάδοση του σήματος ή με την μέτρηση της ισχύς του σήματος. Λόγω της δομής των σύγχρονων κτηρίων και της ανικανότητας των WLAN καρτών δικτύου και των σημείων πρόσβασης, δεν μπορεί να γίνει χρήση και των δύο τιμών στην πράξη.

Η βασική ιδέα είναι να χρησιμοποιηθεί ένα γενικό μοντέλο διάδοσης σήματος και να γίνει παραμετροποίηση του μοντέλου αυτού μετά από κάποιο αριθμό δοκιμαστικών μετρήσεων. Ο μαθηματικός υπολογισμός ενός τέτοιου μοντέλου θα ήταν ιδιαίτερα πολύπλοκος. Η κινητή συσκευή του χρήστη μετράει την ισχύ του σήματος από όλα τα σημεία πρόσβασης που βρίσκονται γύρω του και μεταφέρει τα δεδομένα αυτά σε ένα σύστημα εύρεσης τοποθεσίας το οποίο υπολογίζει τη θέση του χρήστη (EKAHOU positioning engine, [www.ekahau.com](http://www.ekahau.com)).

Μια άλλη προσέγγιση είναι να γίνει τροποποίηση των σημείων πρόσβασης, με σκοπό τον καθορισμό της απόστασης μέχρι την κινητή συσκευή μετρώντας το χρόνο εκτέλεσης του σήματος (signal run time). Συσκευές τέτοιας προσέγγισης κατασκευάζονται από εταιρείες όπως η WhereNet ([www.wherenet.com](http://www.wherenet.com)).

#### 5.2.5 Bluetooth

Το Bluetooth είναι μια τεχνολογία ραδιοσυχνοτήτων κοντινών αποστάσεων, η οποία επιτρέπει την επέκταση της ασύρματης συνδεσιμότητας μεταξύ προσωπικών και επαγγελματικών συσκευών. Η τεχνολογία Bluetooth, παρουσιάζει εμβέλεια της τάξης των 10 μέτρων γύρω από την συσκευή του χρήστη, ο οποίος μπορεί να είναι στατικός ή σε κίνηση. Η βασική ιδέα είναι να δημιουργηθεί ένα δίκτυο ανάμεσα σε συσκευές, όπως αισθητήρες, κάμερες, υπολογιστές χειρός, συσκευές ήχου, κτλ. οι οποίες βρίσκονται κοντά μεταξύ τους και κοντά στον χρήστη.

Η διαδικασία εύρεσης τοποθεσίας με τη χρήση Bluetooth λειτουργεί παρόμοια με αυτή του δικτύου WLAN. Θα περίμενε κανείς ότι εξαιτίας των κοντινών αποστάσεων θα επιτυγχανόταν μεγαλύτερη ακρίβεια, ωστόσο δεν παρέχονται πρότυπες μέθοδοι για την μέτρηση της ισχύς του σήματος.

#### 5.2.6 Άλλες μέθοδοι εύρεσης τοποθεσίας

Στον τομέα της πλοήγησης αεροσκαφών, εμφανίζονται διαφορετικά συστήματα πλοήγησης που βασίζονται σε **φάρους που εκπέμπουν ραδιοκύματα** (radio beacons). Οι φάροι αυτοί μπορούν είτε να χωριστούν με βάση τον τομέα στον οποίον βρίσκονται, είτε να χρησιμοποιούν τη διαφορά φάσης για να καθορίσουν την κατεύθυνση προς αυτούς. Η διαφορά φάσης μεταξύ ενός ζεύγους πομπών, επιτρέπει στους κινητούς σταθμούς να προσδιορίσουν την θέση τους με ακρίβεια από 200 έως 500 μέτρα.

Συμβατικά **συστήματα ραντάρ** χρησιμοποιούνται ευρέως για τον εντοπισμό και την εύρεση τοποθεσίας στη αεροναυπηγία και στη ναυτιλία. Τα νέα συστήματα ραντάρ που υπάρχουν δεν εκπέμπουν το δικό τους σήμα, αλλά χρησιμοποιούν ανακλώμενα σήματα από ραδιοσταθμούς και κινητά τηλέφωνα.

Φυσικά ή τεχνητά **ορόσημα** (landmarks) όπως εγκαταστάσεις ή γεωγραφικοί σχηματισμοί, μπορούν να χρησιμοποιηθούν για την οπτική εύρεση τοποθεσίας, όπως επίσης και η χρήση λέιζερ και τριγωνισμού των περιοχών που περιβάλλουν τα ορόσημα αυτά.

Η τεχνολογία **RFID**, μπορεί να χρησιμοποιηθεί για τον εντοπισμό τοποθεσίας. Πιο συγκεκριμένα οι ετικέτες RFID, βοηθούν στον εντοπισμό αντικειμένων ή ανθρώπων. Ο εντοπισμός αυτός επιτυγχάνεται μέσα σε μικρότερη εμβέλεια από άλλες μεθόδους.

Άλλες συσκευές που μπορούν να χρησιμοποιηθούν για τον προσανατολισμό και τον εντοπισμό στη διαδικασία της πλοήγησης και καθοδήγησης οχημάτων, είναι οι **ψηφιακές πυξίδες**.

Τέλος, υπάρχουν οι **μαγνητικές και οπτικές συσκευές εντοπισμού**. Οι μαγνητικές συσκευές χρησιμοποιούν το μαγνητικό πεδίο για να καθορίσουν τις τρισδιάστατες συντεταγμένες του χρήστη. Οι οπτικές συσκευές κάνουν χρήση των ακτίνων φωτός για να μετρήσουν τις γωνίες που απαιτούνται για να προσδιοριστεί η ζητούμενη θέση.



## 6

## *Αδρανειακό σύστημα*

### *πλοήγησης – Inertial*

### *Navigation System (INS)*

Στο Κεφάλαιο 6, θα περιγραφεί το Αδρανειακό Σύστημα Πλοήγησης – Inertial Navigation System (INS), που αποτελεί την πιο διαδεδομένη μορφή συστήματος πλοήγησης για εσωτερικούς και εξωτερικούς χώρους.

#### **6.1 Περιγραφή συστήματος**

Το INS αποτελεί ένα σύστημα πλοήγησης το οποίο υπολογίζει την παρούσα θέση της συσκευής σε σχέση με την αρχική της τοποθεσία. Αυτό επιτυγχάνεται λαμβάνοντας υπ' όψιν, την επιτάχυνση, την ταχύτητα, την κατεύθυνση και την αρχική θέση της συσκευής. Ένα σύστημα INS συνήθως χρειάζεται ένα επιταχυνσιόμετρο για την μέτρηση της κίνησης, ένα γυροσκόπιο ή κάποια παρόμοια συσκευή για την εύρεση της κατεύθυνσης κίνησης και έναν υπολογιστή για την εκτέλεση των υπολογισμών. Η θέση που είναι σχετική με την αρχική τοποθεσία, μπορεί να υπολογιστεί από τις μετρήσεις του επιταχυνσιόμετρου, οι οποίες παρέχουν πληροφορίες κίνησης σχετικές με προηγούμενα στιγμιότυπα τοποθεσίας. Μόνο με το επιταχυνσιόμετρο, το σύστημα μπορεί να ανιχνεύσει σχετικές κινήσεις, κάτι που κάνει απαραίτητη τη χρήση πυξίδας για τον καθορισμό της κατεύθυνσης κίνησης. [9]

#### **6.2 Dead Reckoning (DR)**

Dead Reckoning (DR), ονομάζεται η διαδικασία κατά την οποία γίνεται ο υπολογισμός της θέσης του αντικειμένου χρησιμοποιώντας ως σημείο αναφοράς την αρχική του θέση. Αυτό επιτυγχάνεται με τον υπολογισμό της παρούσας τοποθεσίας μέσω της εκτιμώμενης ταχύτητας, του χρόνου που χρειάστηκε για να μεταφερθεί από την προηγούμενη θέση στην παρούσα και μέσω της κατεύθυνσης της συσκευής. Τα σύγχρονα INS συστήματα βασίζονται πολύ στην διαδικασία DR και ειδικά οι αυτοματοποιημένες εφαρμογές για την πλοήγηση οχημάτων.

Ένα μειονέκτημα του Dead Reckoning, είναι το ποσοστό των σφαλμάτων, που μπορεί να είναι ιδιαίτερα υψηλό, λόγω της «συσσωρευτικής» φύσης της διαδικασίας. Δηλαδή, για την προσέγγιση της παρούσας θέσης του αντικειμένου είναι απαραίτητη η γνώση της σωστής προηγούμενης θέσης του. Οπότε η όποια πιθανότητα σφάλματος, θα αυξανόταν εκθετικά σε σχέση με το χρόνο. Ένα άλλο πρόβλημα που μπορεί να παρουσιαστεί, είναι ότι παρά την ευρεία χρήση της μεθόδου αυτής στα συστήματα INS, η υλοποίηση της σε ατομικές συσκευές παρουσιάζει αρκετές δυσκολίες λόγω της χαμηλής ποιότητας των αισθητήρων που είναι διαθέσιμοι. Ο θόρυβος του αισθητήρα θα «καλύπτει» το σήμα με αποτέλεσμα την αύξηση της πιθανότητας σφάλματος.

### 6.3 Προσαρμογή στον χάρτη – Map Matching

Η «προσαρμογή στον χάρτη» είναι μία μέθοδος για τη συγχώνευση δεδομένων από την εύρεση τοποθεσίας σήματος και από το δίκτυο του ψηφιακού χάρτη που χρησιμοποιείται για την αναπαράσταση της θέσης (Εικόνα 6.1). Με τη συγχώνευση αυτή, γίνεται η εκτίμηση της θέσης του κινητού αντικειμένου η οποία προσαρμόζεται καλύτερα στον ψηφιακό χάρτη. Η προσαρμογή στον χάρτη, βοηθά περισσότερο όταν η ζητούμενη θέση αναμένεται να βρίσκεται σε ένα συγκεκριμένο μονοπάτι, όπως στο πρόβλημα εντοπισμού ενός κινούμενου οχήματος σε διαδρομή μιας συσκευής GPS.



Εικόνα 6.1 - Map Matching Method

Υπάρχουν δύο ειδών αλγόριθμοι για προσαρμογή στον χάρτη: online και offline. Οι online αλγόριθμοι χρησιμοποιούνται σε περιπτώσεις στις οποίες είναι διαθέσιμα μόνο τα δεδομένα της παρούσας και της προηγούμενης κατάστασης, για την εκτίμηση της θέσης, όπως συμβαίνει και στις συσκευές GPS σε ένα όχημα. Οι offline αλγόριθμοι, χρησιμοποιούνται όταν είναι διαθέσιμες μερικές ή όλες οι μελλοντικές καταστάσεις, όπως για παράδειγμα, μια ήδη γνωστή καταγεγραμμένη διαδρομή ενός κινούμενου αντικειμένου.

### 6.4 Τεχνικές προσαρμογής

Για να είναι χρήσιμη η εικόνα ή το μοντέλο ενός χάρτη, σε εφαρμογή λογισμικού θα πρέπει να γίνεται μετατροπή σε μια καινούργια δομή δεδομένων, η οποία παρέχει την απαραίτητη πληροφορία σε μορφή που είναι προσβάσιμη. Υπάρχουν τέσσερις κατηγορίες στις οποίες επικεντρώνεται η αναπαράσταση ενός χάρτη:

- Συνδεσιμότητα (connectivity)
- Εγγύτητα (proximity)
- Διατομή (intersection)
- Ιδιότητα μέλους (membership)

Διαφορετικές δομές χάρτη μπορεί να επικεντρώνονται σε κάποιες κατηγορίες πιο πολύ, απ' ό,τι σε άλλες. Αν τα αρχικά δεδομένα (raw data) βρίσκονται σε ένα αρχείο CAD, η διαδικασία είναι πιο απλή διότι η δομή έχει ήδη αναλυθεί σε απλά στοιχεία.

Αν το σχέδιο του χάρτη είναι σε μορφή .jpg, .png or .pdf, η διαδικασία για την μετατροπή των αρχικών δεδομένων σε πρωτογενή δομή δεδομένων (primitive data structure), απαιτεί τη χρήση τεχνικών επεξεργασίας εικόνας, συμπεριλαμβανομένου της αναγνώρισης αντικειμένων (object recognition) και φιλτράρισμα δεδομένων (data filtering).

## **6.5 Γραφική αναπαράσταση**

Για την εφαρμογή αλγορίθμων γράφων με σκοπό τον προσδιορισμό του συντομότερου μονοπατιού μεταξύ δύο τοποθεσιών, είναι απαραίτητη η μετατροπή της δομής αναπαράστασης ενός χάρτη, από τοίχους, διαδρόμους και πόρτες σε μια δομή που να διαθέτει μόνο κόμβους και συνδέσεις μεταξύ των κόμβων αυτών (ακμές). Στην γραφική αναπαράσταση, τα δωμάτια, τα σταυροδρόμια, οι σκάλες, οι ανελκυστήρες και άλλα κτηριακά τμήματα αναπαρίστανται από κόμβους, ενώ οι διάδρομοι αναπαρίστανται με συνδέσεις μεταξύ δύο κόμβων που ονομάζονται ακμές. Επίσης, ένας κόμβος διαθέτει πληροφορία σχετική με την τοποθεσία. Η πληροφορία αυτή περιλαμβάνει τις συντεταγμένες του κόμβου και έναν σύνδεσμο προς μια βάση δεδομένων από την οποία ο χρήστης μπορεί να ανακτήσει περισσότερη πληροφορία για τον κόμβο αυτό. Μια σύνδεση μεταξύ δυο κόμβων, δηλώνει την δυνατότητα άμεσης μετάβασης από τον ένα κόμβο στον άλλον και στον γράφο δηλώνονται ως ευθείες γραμμές. Οι καμπύλες και οι στροφές ενός χάρτη, αναπαρίστανται με περισσότερους από έναν συνδέσμους. Οι συμβάσεις σχεδίασης αυτές, έχουν σα σκοπό την πιο αποτελεσματική εκτέλεση του αλγορίθμου διαπέρασης γράφων.

## **6.6 Αλγόριθμος δρομολόγησης**

Σε τελικό στάδιο είναι απαραίτητη η επιλογή ενός αλγορίθμου δρομολόγησης για την εύρεση του συντομότερου μονοπατιού μεταξύ δύο κόμβων σε έναν γράφο. Τυπικά, η σχέση σύνδεσης μεταξύ των κόμβων σε έναν γράφο αναπαρίσταται με έναν πίνακα δύο διαστάσεων. Η σχέση αυτή ονομάζεται βάρος (weight) ή κόστος (cost) για την μετάβαση από τον έναν κόμβο στον άλλον και μπορεί να αντιπροσωπεύει απόσταση, χρόνο ή βαθμό ευκολίας. Η επιλογή της δομής δεδομένων θα επηρεάσει το μέγεθος της βάσης δεδομένων όπως επίσης και την επίδοση του αλγορίθμου δρομολόγησης που θα χρησιμοποιηθεί. Οι διάφοροι αλγόριθμοι εύρεσης συντομότερου μονοπατιού έχουν συζητηθεί και περιγραφεί στο Κεφάλαιο 6.

## **6.7 Συμπεράσματα**

Οι παραπάνω διαδικασίες αποτελούν τα τμήματα από τα οποία δομείται ένα αδρανειακό σύστημα πλοήγησης. Αποτελούν ωστόσο, βασικά δομικά συστατικά οποιουδήποτε συστήματος πλοήγησης τόσο σε εσωτερικούς όσο και εξωτερικούς χώρους. Για κάθε σύστημα μπορεί να δοθεί έμφαση σε κάποιο χαρακτηριστικό πιο πολύ από ένα άλλο σύστημα, με σκοπό την καλύτερη και αποδοτικότερη εξυπηρέτηση των αναγκών του χρήστη. Παρά το γεγονός ότι έχουν προταθεί αρκετά συστήματα πλοήγησης, δεν είναι ακόμα δυνατό να απαλειφθούν οι τεχνικοί και οι χρηστικοί περιορισμοί (χαμηλή ακρίβεια, αναξιοπιστία, δυσκολία στον σχεδιασμό, χαμηλή ασφάλεια και υψηλό κόστος τροποποίησης).

Στο κεφάλαιο αυτό θα γίνει η περιγραφή της εφαρμογής και των λειτουργιών της και θα παρουσιαστεί η αρχιτεκτονική του συστήματος που υλοποιήθηκε.

### 7.1 Εισαγωγή

Στα πλαίσια της παρούσας εργασίας σχεδιάστηκε και υλοποιήθηκε μια εφαρμογή που περιλαμβάνει όλα τα βασικά χαρακτηριστικά της τεχνολογίας RFID. Η εφαρμογή αυτή χρησιμοποιείται για να διαχειριστεί υλικά ή προϊόντα μιας αποθήκης, κάποιας εταιρείας εμφιαλώσεων και εφαρμόστηκε σε έναν κινητό αναγνώστη της iDtronic (Εικόνα 7.1), ο οποίος λειτουργεί σε πολύ υψηλές συχνότητες (UHF ~ 900MHz). Παράλληλα χρησιμοποιήθηκαν παθητικές ετικέτες (passive tags) που χρησιμοποιούν το πρωτόκολλο EPC Gen2, για την αναγνώριση των προϊόντων (Εικόνα 7.2). [7]



Εικόνα 7.1 – Ο αναγνώστης RFID που χρησιμοποιήθηκε (RFID Reader Gun-Type)



Εικόνα 7.2 - Οι παθητικές ετικέτες RFID που χρησιμοποιήθηκαν (RFID passive tag)

## 7.2 Περιγραφή συστήματος

Η εφαρμογή που υλοποιήθηκε ανήκει στην κατηγορία του «πελάτη – εξυπηρετητή» (client - server). Ο πελάτης ζητά από τον εξυπηρετητή πληροφορίες και ο τελευταίος εξυπηρετεί το αίτημα παρέχοντάς του τις πληροφορίες αυτές. Αφού τελειώσει η διαδικασία, ο εξυπηρετητής περιμένει έως ότου κάποιος πελάτης υποβάλλει πάλι κάποια αίτηση για εξυπηρέτηση.

### 7.2.1 Το μοντέλο «πελάτη – εξυπηρετητή»

Στην επιστήμη των ηλεκτρονικών υπολογιστών, το μοντέλο αρχιτεκτονικής «πελάτη - εξυπηρετητή» ή αλλιώς client – server, αποτελεί μία συνήθη μέθοδο ανάπτυξης λογισμικού στην οποία ο πελάτης ή ένα τμήμα λογισμικού (client) ζητά κάτι (π.χ. έναν πόρο, τα αποτελέσματα ενός υπολογισμού, κτλ) και ένα άλλο τμήμα λογισμικού, ο εξυπηρετητής (server), του το επιστρέφει. Κάθε εξυπηρετητής μπορεί να εξυπηρετεί πολλούς πελάτες.

Ο εξυπηρετητής και ο πελάτης μπορούν να εκτελούνται σε διαφορετικές διεργασίες, οι οποίες με τη σειρά τους μπορούν να εκτελούνται σε διαφορετικούς υπολογιστές, οπότε απαιτείται ένα δίκτυο υπολογιστών για την επικοινωνία μεταξύ τους.

### 7.2.2 Από τη μεριά του «πελάτη» (client – side)

Το κύριο κομμάτι της εφαρμογής τρέχει στον αναγνώστη RFID - σε επίπεδο client - και έχει υλοποιηθεί στη γλώσσα προγραμματισμού C# με τη χρήση του προγράμματος ανάπτυξης λογισμικού Microsoft Visual Studio 2008. Επίσης, δημιουργήθηκε μια τοπική βάση δεδομένων που είναι προσπελάσιμη μέσω της γλώσσας προγραμματισμού ειδικού σκοπού SQL (Structured Query Language).

Αρχικά, ο χρήστης αρχίζει μια προσωπική συνεδρία (start session). Έπειτα, σαρώνει με τον αναγνώστη την ετικέτα που βρίσκεται προσκολλημένη σε ένα προϊόν και σαν απάντηση λαμβάνει το EPC του προϊόντος που σάρωσε. Αν το προϊόν αυτό υπάρχει στη βάση δεδομένων, εμφανίζονται οι πληροφορίες που σχετίζονται με αυτό. Σε αντίθετη περίπτωση, ο χρήστης επερωτάται αν επιθυμεί να εισάγει το προϊόν με τις πληροφορίες του στην τοπική βάση δεδομένων και αυτή με τη σειρά της ενημερώνεται.

Ο χρήστης έχει τη δυνατότητα να τροποποιήσει τον ηλεκτρονικό κωδικό προϊόντος (EPC) εάν το επιθυμεί, αλλάζοντας το μοναδικό χαρακτηριστικό που προσδιορίζει το προϊόν.

Κάθε προϊόν είναι αποθηκευμένο σε μια συγκεκριμένη τοποθεσία μέσα στην αποθήκη. Μετά τη σάρωση των προϊόντων η εφαρμογή δίνει τη δυνατότητα στον χρήστη να εμφανίσει το συντομότερο μονοπάτι (shortest path) που περνάει από τις εκάστοτε αποθηκευτικές θέσεις στις οποίες αντιστοιχούν τα προϊόντα, είτε πρόκειται για εναπόθεση μετά από παραλαβή, είτε για συλλογή παραγγελίας και αποστολή.

### 7.2.3 Από τη μεριά του «εξυπηρετητή» (server – side)

Η εφαρμογή από τη μεριά του εξυπηρετητή, έχει αναπτυχθεί στη γλώσσα προγραμματισμού PHP και τρέχει σε έναν WAMP server. Το WAMP είναι ένα πακέτο που εγκαθιστά εύκολα και γρήγορα τα απαραίτητα προγράμματα που απαιτούνται από το λειτουργικό των Microsoft Windows, για να εγκατασταθεί ένας τοπικός web server. Η ονομασία του προέρχεται από τις λέξεις Windows, Apache, MySQL, PHP. Apache είναι το λειτουργικό που χρειάζεται για να «τρέξει» ο εξυπηρετητής, MySQL είναι ένα σύστημα βάσεων δεδομένων μπορούν να αποθηκευτούν πληροφορίες, PHP είναι η γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο.

Όταν ο χρήστης τελειώσει τη συνεδρία του (end session), πρέπει να ενημερώσει την απομακρυσμένη βάση δεδομένων (που βρίσκεται σε κάποιον υπολογιστή) για τυχόν νέα προϊόντα που εισήγαγε σε αυτήν. Για να γίνει αυτό θα πρέπει να υπάρχει πρόσβαση στο διαδίκτυο.

Ο λόγος που χρησιμοποιούνται δύο βάσεις δεδομένων είναι ότι, σε μεγάλες εγκαταστάσεις ή εργοστάσια, μπορεί να υπάρχει παντού κάλυψη ασύρματου δικτύου (Wi-Fi), κάτι που θα έκανε αδύνατη την επικοινωνία με έναν απομακρυσμένο server. Έτσι, σε πρώτο στάδιο, ο χρήστης ενημερώνει την τοπική βάση δεδομένων που βρίσκεται στον αναγνώστη RFID και στη συνέχεια όταν εισέλθει σε εμβέλεια δικτύου, ενημερώνει και την βάση δεδομένων του web server.

## 7.3 Αρχιτεκτονική συστήματος

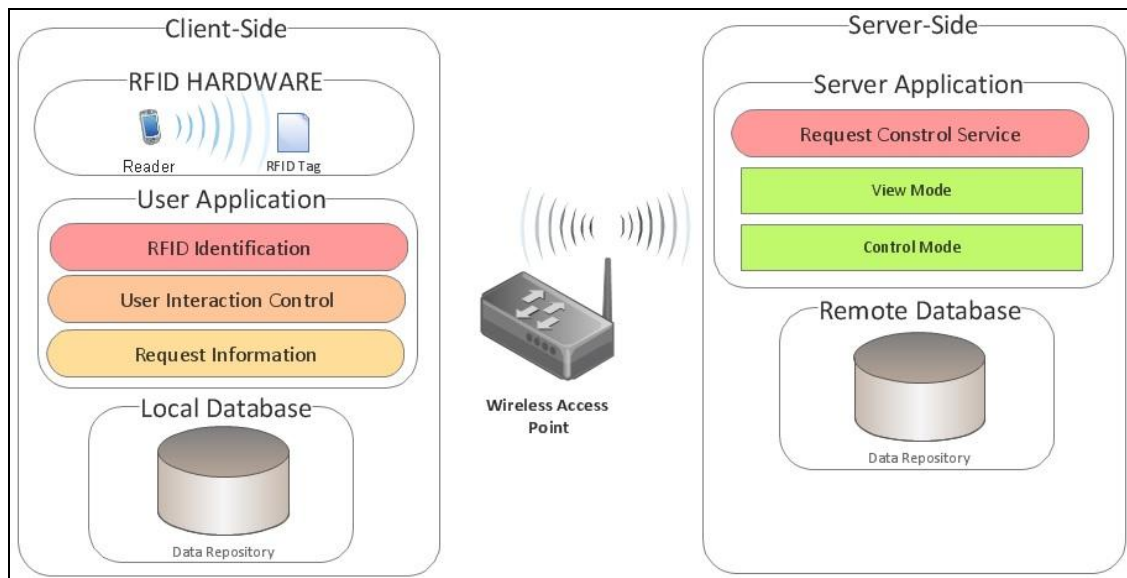
Η αρχιτεκτονική του συστήματος σε επίπεδο client, αποτελείται από τρία τμήματα και μία τοπική βάση δεδομένων:

- i. Αναγνώριση RFID (RFID Identification module): χρησιμοποιεί μια .NET βιβλιοθήκη που παρέχεται από τον κατασκευαστή του PDA για τη διαμόρφωση και την επερώτηση του αναγνώστη RFID.
- ii. Έλεγχος διεπαφής χρήστη (User Interface Control module): αποτελεί τον πυρήνα της client εφαρμογής, υλοποιεί τη λογική της και χειρίζεται τη διεπαφή χρήστη.
- iii. Αίτημα πληροφορίας (Request Information module): χρησιμοποιεί τα δεδομένα που ανακτήθηκαν από την ανάγνωση της ετικέτας RFID και διαμορφώνει τα αιτήματα δεδομένων (data requests) προς τον εξυπηρετητή. Τα αιτήματα δεδομένων έχουν τη μορφή υπέρ-συνδέσμων (hyperlinks) και οι απαντήσεις (replies) λαμβάνονται σε html.
- iv. Τοπική βάση δεδομένων (Local Database): βρίσκεται σε έναν τοπικό εξυπηρετητή SQL, που χρησιμοποιείται σε κινητές συσκευές (SQL Server Compact Edition) και αποθηκεύεται στη συσκευή αναγνώστη, σε ένα αρχείο μορφής spatial data file (.sdf).

Η αρχιτεκτονική του συστήματος σε επίπεδο server, αποτελείται από ένα τμήμα εξυπηρέτησης αιτήματος (**Request Control Service**), που είναι υπεύθυνο για την εύρεση και παρουσίαση των δεδομένων που ζητήθηκαν και από μία απομακρυσμένη (σε σχέση με τον client) βάση δεδομένων αναπτυγμένη σε έναν MySQL server. Το τμήμα εξυπηρέτησης αιτήματος, αποτελείται από δύο υπό-τμήματα:

- i. Τμήμα ελέγχου (**Control module**): για την πρόσβαση στα δεδομένα της βάσης
- ii. Τμήμα όψης (**View module**): για την αναπαράσταση των δεδομένων αυτών

Η αρχιτεκτονική του συστήματος εμφανίζεται παρακάτω, στην Εικόνα 7.3:

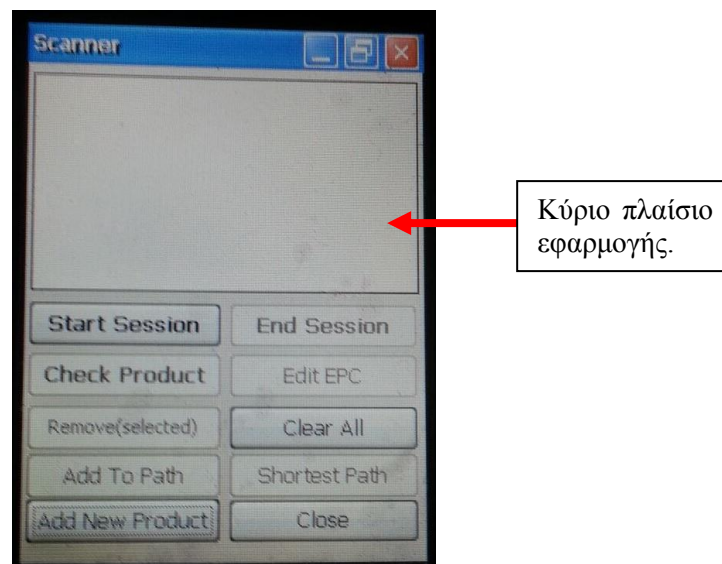


*Εικόνα 7.3 - Αρχιτεκτονική συστήματος*

Στο Κεφάλαιο 8 θα παρουσιαστεί η εφαρμογή που υλοποιήθηκε, με όλα τα χαρακτηριστικά και τις δυνατότητες που παρέχει στο χρήστη. Θα δοθούν επίσης, και ενδεικτικά κομμάτια κώδικα για την καλύτερη αντίληψη της υλοποίησης.

### 8.1 Η φόρμα “Scanner”

Πρόκειται για την αρχική φόρμα της εφαρμογής που εμφανίζεται με την εκτέλεσή της (Εικόνα 8.1).



Εικόνα 8.1 - Η αρχική φόρμα Scanner

Η φόρμα “Scanner” αποτελείται από ένα κύριο πλαίσιο, στο οποίο εμφανίζονται οι κωδικοί των ετικετών που σαρώθηκαν και από πέντε κουμπιά – επιλογές :

- 1) **Start Session:** έναρξη συνεδρίας για έναν χρήστη
- 2) **End Session:** τερματισμός συνεδρίας
- 3) **Check Product:** έλεγχος διαθεσιμότητας προϊόντος
- 4) **Edit EPC:** τροποποίηση ηλεκτρονικού κωδικού προϊόντος από τον χρήστη



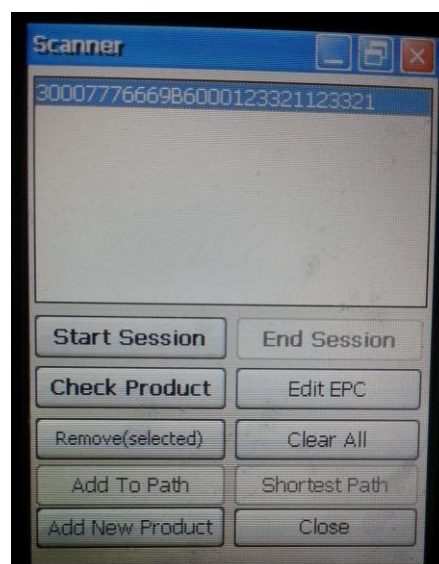
- 5) **Remove (Selected):** αφαίρεση επιλεγμένου ηλεκτρονικού κωδικού προϊόντος από το κεντρικό πλαίσιο σάρωσης
- 6) **Clear All:** καθαρισμός κύριου πλαισίου εφαρμογής
- 7) **Add to Path:** εισαγωγή προϊόντος στη διαδικασία εύρεσης του συντομότερου μονοπατιού
- 8) **Shortest Path:** υπολογισμός και εμφάνιση του συντομότερου μονοπατιού
- 9) **Add New Product:** εισαγωγή στοιχείων νέου προϊόντος
- 10) **Close:** κλείσιμο/τερματισμός της εφαρμογής

Όπως φαίνεται στην Εικόνα 8.1, από τα παραπάνω κουμπιά, μόνο τα τέσσερα από αυτά είναι ενεργοποιημένα κατά την έναρξη της εφαρμογής: Start Session, Add New Product, Clear All και Close.

Αρχικά ο χρήστης πατάει το κουμπί Start Session για να αρχίσει τη συνεδρία. Κάθε φορά που γίνεται εισαγωγή ενός νέου προϊόντος στην τοπική βάση δεδομένων, αποθηκεύεται η εισαγωγή αυτή, ούτως ώστε να ενημερωθεί και η απομακρυσμένη βάση δεδομένων στο τέλος της συνεδρίας του χρήστη, δηλαδή, όταν πατήσει το κουμπί End Session. Το πάτημα του κουμπιού Start Session προκαλεί την απενεργοποίηση του κουμπιού End Session και αντίστροφα.

Η εισαγωγή ενός προϊόντος, μπορεί να γίνει με δύο τρόπους. Είτε πατώντας το κουμπί Add New Product και ο χρήστης μεταφέρεται στη φόρμα εισαγωγής πληροφοριών προϊόντος, είτε με τη σάρωση της ετικέτας και αφού διαπιστωθεί ότι το προϊόν δεν υπάρχει στη βάση δεδομένων. Στην περίπτωση αυτή ο χρήστης ερωτάται αν επιθυμεί να εισάγει το νέο προϊόν στην τοπική βάση δεδομένων και αν ο χρήστης απαντήσει καταφατικά, εμφανίζεται η αντίστοιχη φόρμα εισαγωγής πληροφοριών προϊόντος.

Για να αναγνωριστεί ένα προϊόν, θα πρέπει να γίνει σάρωση της ετικέτας του. Αυτό επιτυγχάνεται με το πάτημα της σκανδάλης του RFID Gun Reader. Όταν γίνει επιτυχής σάρωση μιας ετικέτας εμφανίζεται ο ηλεκτρονικός κωδικός προϊόντος στο κύριο πλαίσιο εφαρμογής και ακούγεται στη συσκευή ένας χαρακτηριστικός ήχος επιτυχίας (Εικόνα 8.2).

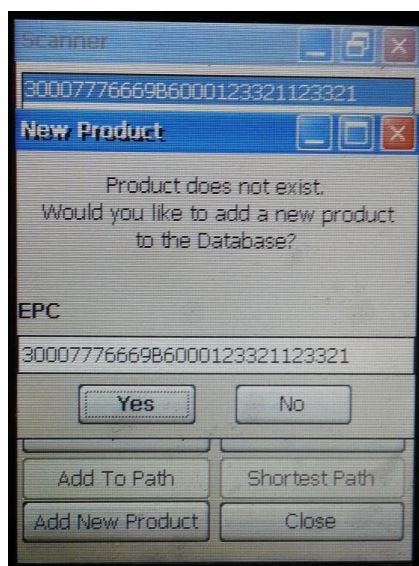


Εικόνα 8.2 - Αναγνώριση ετικέτας με το πάτημα της σκανδάλης

Με την επιτυχή αναγνώριση μιας ετικέτας, ενεργοποιούνται τα κουμπιά: Remove (Selected), Edit EPC και Check Product. Επιλέγοντας έναν κωδικό από το κύριο πλαίσιο και πατώντας Remove (Selected), αφαιρείται ο συγκεκριμένος κωδικός από τη λίστα.

## 8.2 Η φόρμα “New Product”

Πατώντας το κουμπί Check Product γίνεται έλεγχος για τη διαθεσιμότητα του προϊόντος στην τοπική βάση δεδομένων. Σε περίπτωση που ο κωδικός δεν υπάρχει στη βάση, εμφανίζεται η φόρμα “New Product”. Η φόρμα αυτή ενημερώνει τον χρήστη για την μη ύπαρξη του συγκεκριμένου προϊόντος και τον ρωτάει εάν θέλει να το εισάγει στη βάση δεδομένων, όπως φαίνεται στην Εικόνα 8.3:



Εικόνα 8.3 - Η φόρμα “New Product”

Αν ο χρήστης πατήσει το κουμπί No, γίνεται επιστροφή στη φόρμα “Scanner”, ενώ αν πατήσει το Yes, μεταφέρεται στη φόρμα εισαγωγής πληροφοριών προϊόντος “Add Product Info”.

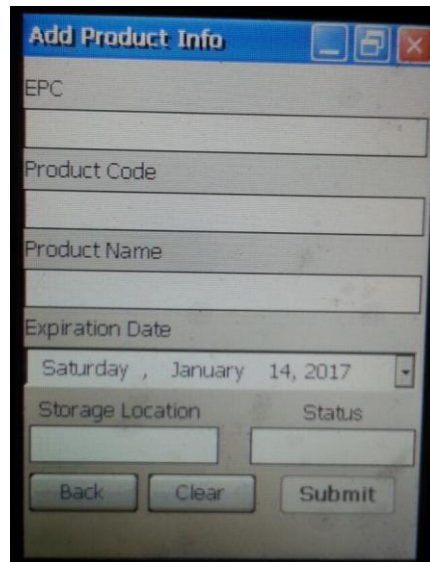
## 8.3 Η φόρμα “Add Product Info”

Στη φόρμα εισαγωγής πληροφοριών προϊόντος “Add Product Info” (Εικόνα 8.4) υπάρχουν έξι πεδία στα οποία ο χρήστης εισάγει τα χαρακτηριστικά του προϊόντος. Αυτά είναι:

- 1) **EPC:** ο ηλεκτρονικός κωδικός προϊόντος της ετικέτας
- 2) **Product Code:** ο κωδικός αναγνώρισης προϊόντος της εταιρείας
- 3) **Product Name:** το όνομα του προϊόντος
- 4) **Expiration Date:** η ημερομηνία λήξης του προϊόντος
- 5) **Storage Location:** η τοποθεσία αποθήκευσης του προϊόντος
- 6) **Status:** η κατάσταση υγείας του προϊόντος (υγιές ή ληγμένο)

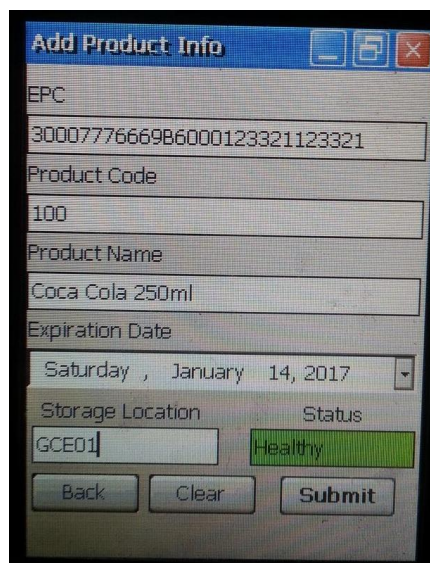
Η φόρμα διαθέτει και τρία κουμπιά – επιλογές:

- 1) **Back:** κλείσιμο της φόρμας και επιστροφή στην αρχική φόρμα “Scanner”
- 2) **Clear:** καθαρισμός όλων των πεδίων της φόρμας
- 3) **Submit:** καταχώρηση του νέου προϊόντος στη βάση δεδομένων (αρχικά είναι απενεργοποιημένο)

A screenshot of a web application window titled "Add Product Info". The window contains several input fields: "EPC", "Product Code", "Product Name", "Expiration Date" (with a date picker set to "Saturday, January 14, 2017"), "Storage Location", and "Status". At the bottom, there are three buttons: "Back", "Clear", and "Submit". The "Submit" button is currently disabled.

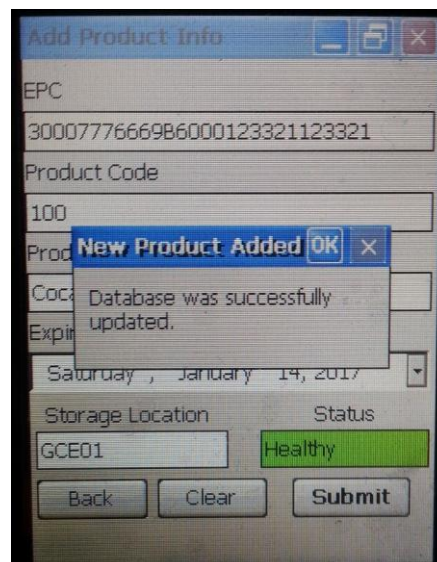
*Εικόνα 8.4 - Η φόρμα "Add Product Info"*

Όταν συμπληρωθούν όλα τα πεδία ενεργοποιείται το κουμπί Submit. Επίσης, όταν δοθεί η ημερομηνία λήξης του προϊόντος, αλλάζει το χρώμα του πεδίου Status (πράσινο για υγιές προϊόν, κόκκινο για ληγμένο προϊόν) και συμπληρώνεται αυτόματα με μια από τις επιλογές Healthy ή Expired (Εικόνα 8.5).

A screenshot of the same "Add Product Info" form, but now with data entered. The "EPC" field contains "3000777666986000123321123321", "Product Code" contains "100", "Product Name" contains "Coca Cola 250ml", and "Expiration Date" is still "Saturday, January 14, 2017". The "Storage Location" field contains "GCE01". The "Status" field is now green and contains the text "Healthy". The "Submit" button is now enabled.

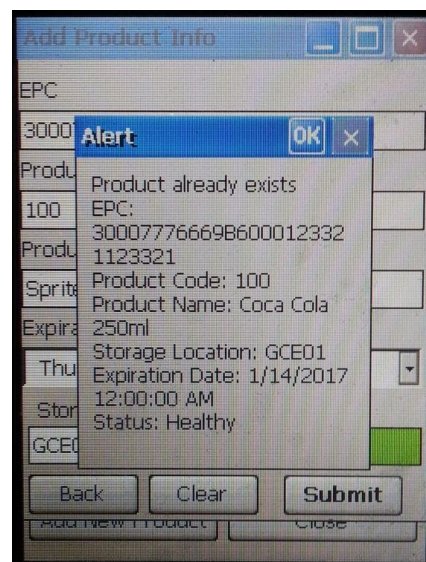
*Εικόνα 8.5 - Η φόρμα "Add Product Info" με συμπληρωμένα πεδία*

Με το πάτημα του κουμπιού Submit, γίνεται έλεγχος της εγκυρότητας των πεδίων και στη συνέχεια ενημερώνεται η τοπική βάση δεδομένων, εμφανίζοντας ένα μήνυμα επιτυχούς καταχώρησης (Εικόνα 8.6).



*Εικόνα 8.6 - Επιτυχής καταχώρηση*

Μήνυμα σφάλματος εμφανίζεται στις περιπτώσεις που στο πεδίο EPC δεν έχουν εισαχθεί 28 χαρακτήρες ή δεν έχει δοθεί σωστή τοποθεσία αποθήκευσης στο πεδίο Storage Location. Επίσης, εμφανίζεται μήνυμα ειδοποίησης σε περίπτωση που εισαχθεί προϊόν με το ίδιο EPC ή Product Code (Εικόνα 8.7).

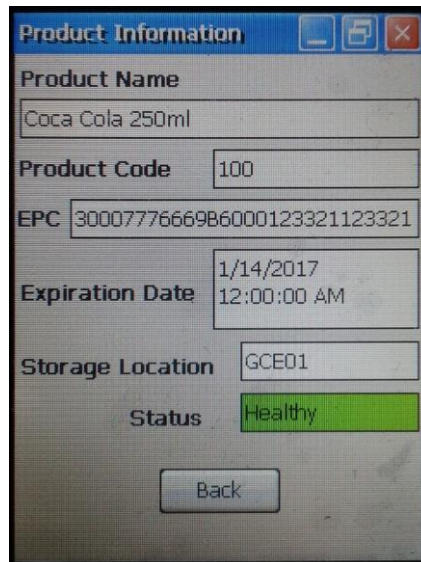


*Εικόνα 8.7 - Μήνυμα για ήδη υπάρχον προϊόν*



## 8.4 Η φόρμα “Product Information”

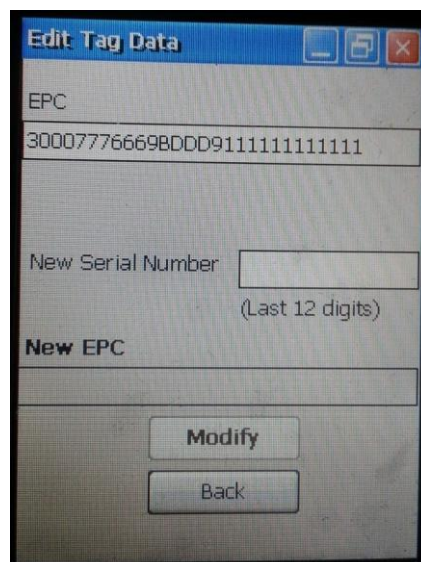
Αν μετά το πάτημα του κουμπιού Check Product της φόρμας “Scanner”, διαπιστωθεί ότι το EPC υπάρχει στη βάση δεδομένων, τότε εμφανίζεται η φόρμα “Product Information” με τις πληροφορίες του προϊόντος. Τα πεδία της φόρμας αυτής είναι μόνο για ανάγνωση (read-only), δηλαδή δεν μπορεί να εισαχθεί ή να τροποποιηθεί πληροφορία (Εικόνα 8.8). Με το πάτημα του κουμπιού Back, γίνεται επιστροφή στην φόρμα “Scanner”.



Εικόνα 8.8 - Η φόρμα “Product Information”

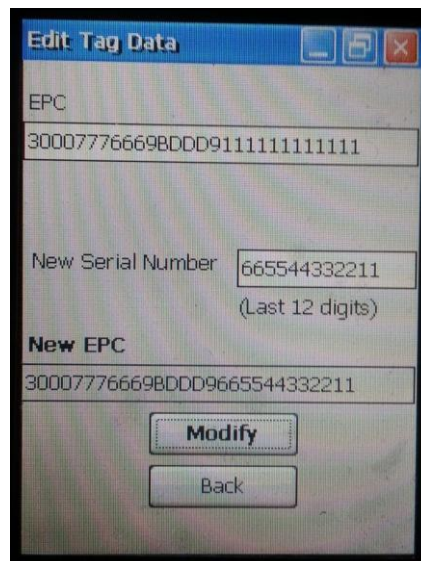
## 8.5 Η φόρμα “Edit Tag Data”

Η εφαρμογή παρέχει στο χρήστη τη δυνατότητα να τροποποιήσει το EPC. Αφού επιλεγεί το EPC προς τροποποίηση από το κύριο πλαίσιο και πατηθεί το κουμπί Edit EPC γίνεται μεταφορά στη φόρμα “Edit Tag Data” (Εικόνα 8.9).



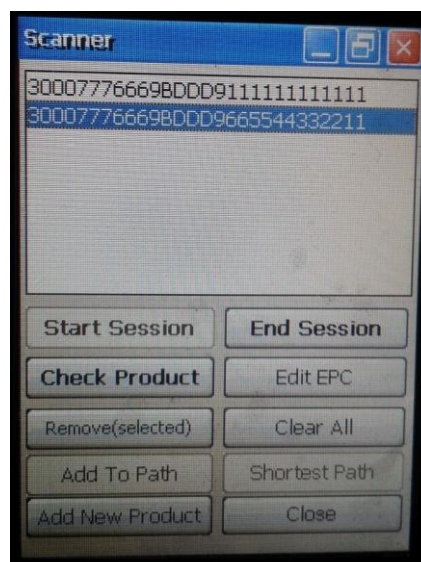
Εικόνα 8.9 - Η φόρμα “Edit Tag Data”

Στη φόρμα αυτή εμφανίζονται τρία πεδία: EPC, New Serial Number και New EPC. Από αυτά το EPC και New EPC είναι μόνο για ανάγνωση. Στο πεδίο EPC εμφανίζεται ο κωδικός της ετικέτας που πρόκειται να τροποποιηθεί. Στη συνέχεια ο χρήστης εισάγει τα 12 τελευταία ψηφία του νέου κωδικού και στο πεδίο New EPC εμφανίζεται το νέο EPC που προκύπτει. Το κουμπί Modify ενεργοποιείται όταν εισαχθούν 12 ψηφία στο πεδίο New Serial Number. Το κουμπί Back προκαλεί μετάβαση στη φόρμα “Scanner” (Εικόνα 8.10).



*Εικόνα 8.10 - Τροποποίηση του EPC*

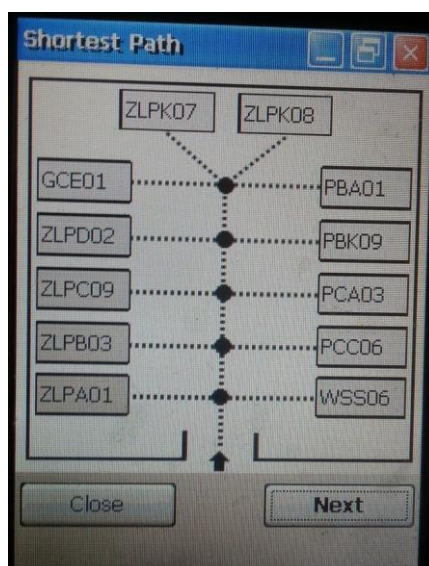
Με το πάτημα του κουμπιού Modify γίνεται η αλλαγή του ηλεκτρονικού κωδικού προϊόντος και η ετικέτα αποκτά το νέο της EPC. Έτσι, όταν σαρωθεί πάλι η ετικέτα θα εμφανιστεί το τροποποιημένο EPC (Εικόνα 8.11).



*Εικόνα 8.11 - Σάρωση της ίδιας ετικέτας πριν και μετά την αλλαγή EPC*

## 8.6 Η φόρμα “Shortest Path”

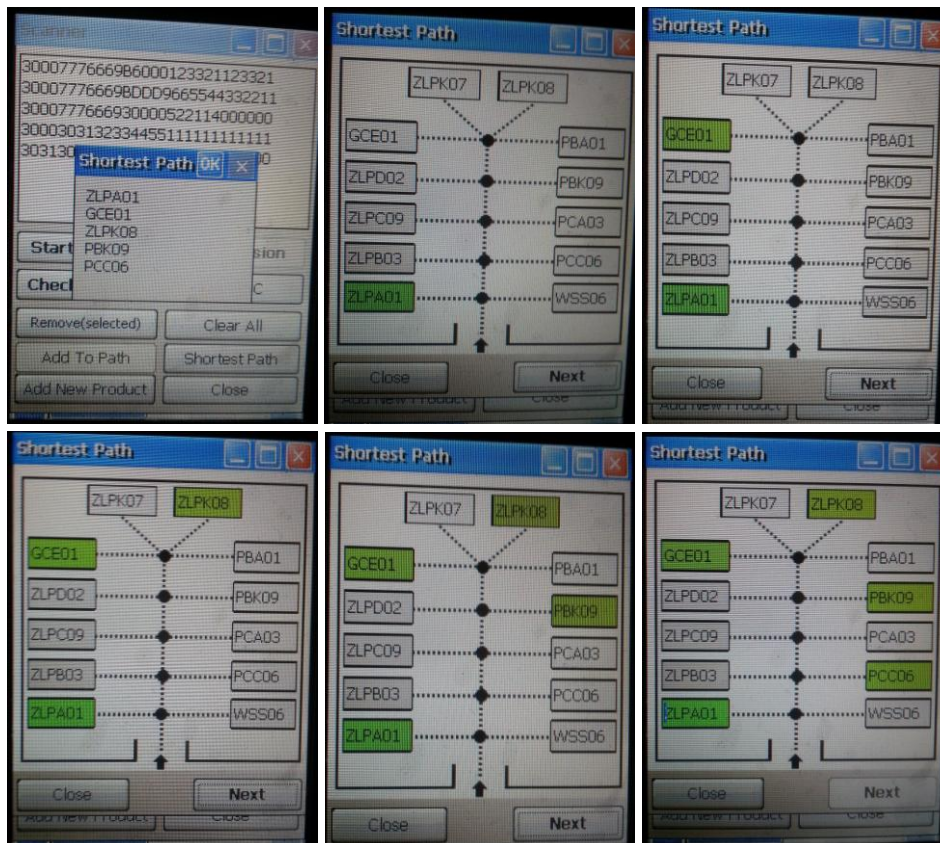
Για κάθε προϊόν που σαρώνεται και υπάρχει στη βάση δεδομένων, ο χρήστης μπορεί να το εισάγει στη μέθοδο υπολογισμού του συντομότερου μονοπατιού, επιλέγοντας το EPC του από το κύριο πλαίσιο και πατώντας το κουμπί Add To Path. Έτσι, όταν ο χρήστης πατήσει το κουμπί Shortest Path, θα γίνει ο υπολογισμός του συντομότερου μονοπατιού, με βάση τις αποστάσεις των τοποθεσιών της αποθήκης στις οποίες είναι αποθηκευμένα τα προϊόντα που έχουν επιλεγεί. Μετά τον υπολογισμό του συντομότερου μονοπατιού, εμφανίζεται η φόρμα “Shortest Path”, όπου παρουσιάζεται η κάτοψη μιας υποτιθέμενης αποθήκης στην οποία φαίνονται όλες οι τοποθεσίες αποθήκευσης μαζί με τα αναγνωριστικά τους (Εικόνα 8.12).



Εικόνα 8.12 - Η φόρμα "Shortest Path"

Πατώντας το κουμπί Next σηματοδοτείται, με πράσινο χρώμα, η επόμενη τοποθεσία στην οποία πρέπει να μεταβεί ο χρήστης, με βάση το συντομότερο μονοπάτι των τοποθεσιών που επέλεξε. Το κουμπί Close κλείνει τη φόρμα “Shortest Path”.

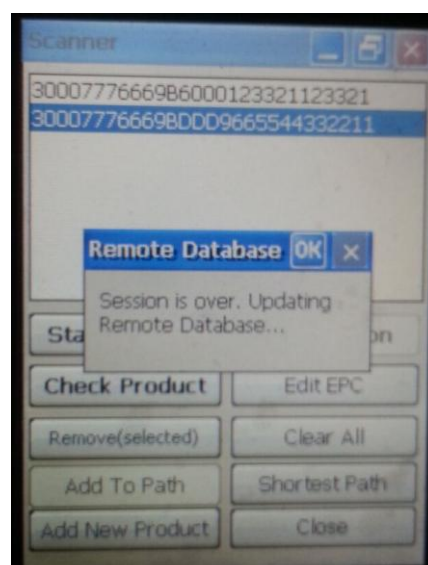
Έστω ότι έγινε η σάρωση πέντε ετικετών και προέκυψαν οι θέσεις αποθήκευσης με την παρακάτω σειρά: GCE01, PBK09, ZLPA01, ZLPK08, PCC06. Για κάθε προϊόν που σαρώθηκε ο χρήστης πατάει το κουμπί Add To Path. Όταν σαρωθεί και το πέμπτο προϊόν, ο χρήστης πατάει το κουμπί Shortest Path. Αφού εκτελεστεί η συνάρτηση εύρεσης συντομότερου μονοπατιού, εμφανίζεται η φόρμα “Shortest Path” και πατώντας το κουμπί Next, ο χρήστης βλέπει την τοποθεσία που πρέπει να επισκεφθεί κάθε φορά, μέχρις ότου να καλυφθούν και οι πέντε θέσεις των προϊόντων που σαρώθηκαν. Ένα στιγμιότυπο εμφάνισης συντομότερου μονοπατιού, φαίνεται παρακάτω στην Εικόνα 8.13:



Εικόνα 8.13 - Στιγμιότυπο εύρεσης συντομότερου μονοπατιού

## 8.7 Η φόρμα “Navigation”

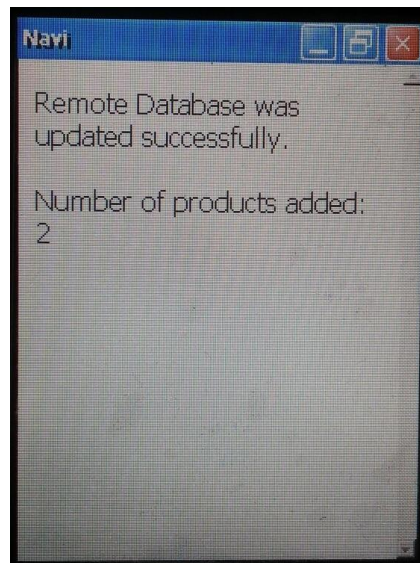
Όταν ο χρήστης ολοκληρώσει τη συνεδρία του, πατάει το κουμπί End Session και εμφανίζεται ανάλογο μήνυμα για ενημέρωση της απομακρυσμένης βάσης δεδομένων (Εικόνα 8.14).



Εικόνα 8.14 - Τερματισμός συνεδρίας



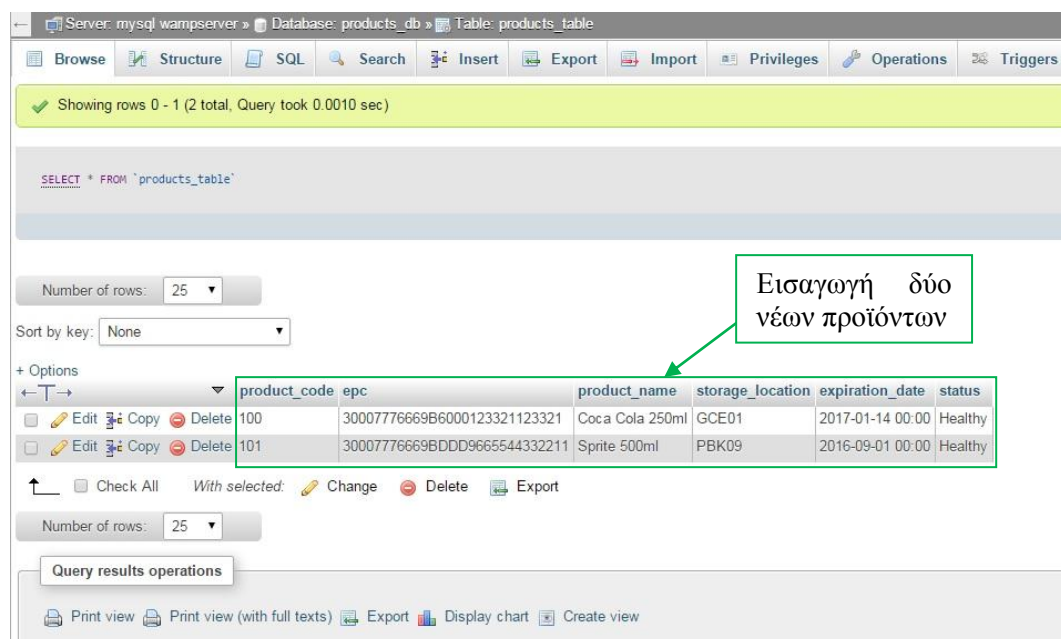
Στη συνέχεια ενημερώνεται η απομακρυσμένη βάση δεδομένων με όσες εισαγωγές προϊόντων έκανε ο χρήστης κατά τη διάρκεια της συνεδρίας του και εμφανίζεται η φόρμα "Navigation" με ανάλογο μήνυμα για την επιτυχημένη ή αποτυχημένη ολοκλήρωση της διαδικασίας ενημέρωσης (Εικόνα 8.15).



Εικόνα 8.15 - Η φόρμα "Navigation"

## 8.8 Η απομακρυσμένη βάση δεδομένων

Ο διαχειριστής της απομακρυσμένης βάσης δεδομένων έχει τη δυνατότητα να δει όλες τις εισαγωγές που έχουν γίνει από τον χρήστη (Εικόνα 8.16). Παράλληλα μπορεί να επεξεργαστεί τις εγγραφές της βάσης και να κάνει τροποποιήσεις τόσο στα δεδομένα όσο και στους πίνακες, με ενέργειες όπως: SELECT, INSERT, ALTER, DELETE, DROP, κτλ.



Εικόνα 8.16 - Ενημέρωση της απομακρυσμένης βάσης δεδομένων

## 8.9 Ενδεικτικός κώδικας υλοποίησης σε C#, SQL και PHP

- Για την λειτουργία του RFID Gun Reader, χρησιμοποιήθηκε το SDK (Software Development Kit) της συσκευής. Πρόκειται για μια βιβλιοθήκη που ονομάζεται AT570UHF\_NET.dll και χρησιμοποιείται με την εντολή:

```
using AT570UHF_NET;
```

- Η δημιουργία αντικειμένων της κλάσης AT570UHF\_NET γίνεται με την εντολή:

```
public static AT570UHF_NET.AT570UHF_NET RFID;  
RFID = new AT570UHF_NET.AT570UHF_NET();
```

- Για την ενεργοποίηση του αναγνώστη, και την ανάγνωση των ετικετών RFID, χρησιμοποιήθηκαν οι μέθοδοι:

```
RFID.AT570UHF_PowerON();  
RFID.AT570UHF_HWND();  
RFID.AT570UHF_Open();
```

```
RFID.GetMemoryData += new  
AT570UHF_NET.AT570UHF_NET.UHF_GET_MEMORY_DATAEventHandler(RFID_  
GetMemoryData);
```

```
RFID.GetReply += new  
AT570UHF_NET.AT570UHF_NET.UHF_GET_REPLYEventHandler(RFID_GetRep  
ly);
```

- Με το πάτημα της σκανδάλης καλείται η συνάρτηση:

```
protected override void OnKeyDown(KeyEventArgs e)
```

η οποία με τη σειρά της καλεί τις μεθόδους για την ανάγνωση του EPC της ετικέτας και την αναπαραγωγή του ήχου ειδοποίησης:

```
RFID.AT570UHF_ReadUID(AT570UHF_NET.AT570UHF_NET.UIDREAD_CODE.EP  
C_GEN2_ONE_TAG);  
PlaySuccess();
```

- Για την δημιουργία και την πρόσβαση στην τοπική βάση δεδομένων χρησιμοποιήθηκαν τα namespaces, SqlServerCe και SqlClient:

```
using System.Data.SqlServerCe;  
using System.Data.SqlClient;
```

Η σύνδεση με την τοπική βάση δεδομένων έγινε με τη δημιουργία αντικειμένου:

```
SqlCeConnection sql_con = new SqlCeConnection("Data  
Source=\\some_file.sdf");
```

με το οποίο γίνεται σύνδεση με το αρχείο (.sdf) της τοπικής βάσης δεδομένων.

- Για την εκτέλεση εντολών SQL δημιουργήθηκαν αντικείμενα της μορφής:

```
SqlCeCommand sql_command = new SqlCeCommand("SELECT COUNT(*)  
FROM [products_table] WHERE ([epc] = @epc)", sql_con);
```

- Για την ανάγνωση των αποτελεσμάτων της εκτέλεσης των SQL εντολών στην τοπική βάση δεδομένων, δημιουργήθηκαν αντικείμενα της μορφής:

```
SqlCeDataReader sql_reader;
```

- Για την τροποποίηση του EPC χρησιμοποιήθηκε η μέθοδος:

```
RFID.AT570UHF_BankSelectWrite(AT570UHF_NET.AT570UHF_NET.MEMBANK_CODE.BANK_EPC, start, WriteData);
```

- Για την εύρεση του συντομότερου μονοπατιού, υλοποιήθηκε ο αλγόριθμος Dijkstra που αποτελεί μέθοδο (shortest\_path) της κλάσης Graph:

```
Graph g = new Graph();
public List<char> shortest_path(char start, char finish){}
```

- Ο ορισμός των αποστάσεων μεταξύ των τοποθεσιών αποθήκευσης έγινε με την κλήση της μεθοόδου:

```
Dictionary<char, Dictionary<char, int>> vertices = new Dictionary<char, Dictionary<char, int>>();
g.add_vertex('C', new Dictionary<char, int>()
{{ 'A', 5 }, { 'B', 8 }, { 'E', 12 }, { 'G', 2 }});
```

- Και η αντιστοίχιση με την ονομασία της τοποθεσίας:

```
foreach (char ch in nodes_shorted)
{
    switch (ch)
    {
        ...
        case 'C':
            final_list.Add("ZLPC09");
            break;
        ...
        default:
            MessageBox.Show("Invalid Location Detected",
                "Alert");
            break;
    }
}
```

- Για την πρόσβαση στην απομακρυσμένη βάση δεδομένων χρησιμοποιήθηκε το αντικείμενο και οι μέθοδοι:

```
System.Uri uri;

uri = new Uri("http://192.168.2.4/check_epc.php?code=" +
Program.insertions[i][0]+...);

nav = new Navi();
nav.webBrowser1.Navigate(uri);
nav.Show();
```

- Ο κώδικας PHP για την απομακρυσμένη βάση δεδομένων:

```
<html>
<head>
</head>
<body>

<?php

    $code = $_GET["code"];
    $epc = $_GET["epc"];
    $name = $_GET["name"];
    $location = $_GET["location"];
    $exp = date($_GET["exp"]);
    $status = $_GET["status"];
    $inserts = $_GET["inserts"];

    $con = mysqli_connect("localhost","username","password");

    if (!$con){
        die("Could not connect: " . mysql_error());
    }

    mysqli_select_db($con,"products_db");

    $query = "INSERT INTO products_table (product_code,
    epc, product_name, storage_location, expiration_date,
    status)
    VALUES ('$code', '$epc', '$name', '$location', '$exp',
    '$status')";

    if (mysqli_query($con,$query))
    {
        echo "Remote Database was updated successfully.";
        echo "<br />";
        echo "Number of products added: ". $inserts;
    }
    else
    {
        echo "Error: " . $query . "<br />" .
        mysqli_error($con);
    }
    mysqli_close($con);
?>
</body>
</html>
```

Στην παρούσα διπλωματική εργασία έγινε μια προσπάθεια ανάλυσης και προσέγγισης της τεχνολογίας RFID. Παρουσιάστηκαν τα βασικά της χαρακτηριστικά και τα στοιχεία από τα οποία αποτελείται ένα τέτοιο σύστημα. Στα πλαίσια της εργασίας, σχεδιάστηκε και υλοποιήθηκε μια εφαρμογή για PDA, που εκμεταλλεύεται τις λειτουργίες της τεχνολογίας RFID, με σκοπό την διευκόλυνση της αναγνώρισης αντικειμένων στον κλάδο του εμπορίου.

Η μέθοδοι αναγνώρισης αντικειμένων από απόσταση έχουν ήδη καθιερωθεί σε πολλές από τις δραστηριότητες του ανθρώπου, γεγονός που σημαίνει ότι οι μέθοδοι αυτοί επιδέχονται βελτιστοποίηση και τροποποιήσεις. Τα συστήματα αναγνώρισης εξελίσσονται και αναπτύσσονται με γρήγορους ρυθμούς τα τελευταία χρόνια και όλο και πιο πολύ φορείς και επιχειρήσεις στρέφουν το ενδιαφέρον τους στις τεχνολογίες αυτές.

Στα πλαίσια της εργασίας, έγινε η δοκιμή συνδυασμού της τεχνολογίας RFID, με τον ήδη υπό εξέλιξη τομέα της πλοήγησης και πιο συγκεκριμένα, της εύρεσης συντομότερου μονοπατιού σε ένα δίκτυο. Παρουσιάστηκαν τα συστήματα και οι μέθοδοι που χρησιμοποιούνται για την εύρεση τοποθεσίας σε εσωτερικούς και εξωτερικούς χώρους και έγινε περιγραφή των κυριότερων αλγορίθμων εύρεσης συντομότερου μονοπατιού.

Ο συνδυασμός των μεθόδων αυτών και των αλγορίθμων εύρεσης συντομότερου μονοπατιού οδήγησε σε ένα πρότυπο σύστημα πλοήγησης, γνωστό και ως «Αδρανειακό Σύστημα Πλοήγησης» (Inertial Navigation System) το οποίο και παρουσιάστηκε στην παρούσα εργασία.

Στην ήδη αναπτυσσόμενη τεχνολογία της αναγνώρισης από απόσταση, η προσθήκη τεχνολογιών από τον τομέα της πλοήγησης θα μπορούσε να βοηθήσει στη δημιουργία συστημάτων που θα απλοποιούσαν κατά πολύ διαδικασίες εύρεσης, αναγνώρισης και προσπέλασης αντικειμένων ή ανθρώπων.

- [1] *Radio Frequency Identification*. <https://goo.gl/oQsK63>
- [2] **Athanasios Fevgas, Nikolaos Fraggogiannis, Panagiota Tsompanopoulou, and Panayiotis Bozanis**, Department of Electrical & Computer Engineering, University of Thessaly. *The iMuse Virtual Museum: towards a cultural education platform*, July 2014
- [3] *The History of RFID technology*. <http://www.rfidjournal.com/articles/view?1338/>
- [4] *The Different types of RFID systems*. <http://goo.gl/kZifVv>
- [5] *How to use RFID technology*. <http://goo.gl/Km4AWF>
- [6] *Understanding the EPC Gen 2 Protocol*. <https://goo.gl/rxe5CR>
- [7] **IDtronic**, UHF 900 MHz .NET SDK API Reference and User Manual Guide
- [8] **Bill R, Cap C, Kofahl M and Mundt T**, University of Rostock Germany. *Indoor and Outdoor Positioning in Mobile environments – a review and some Investigations on WLAN – Positioning*, Vol.10, No.2, December 2004
- [9] **Manh Hung V. Le, Dimitris Saragas, Nathan Webb**, Worcester Polytechnic Institute, Massachusetts, USA. *Indoor Navigation System for Handheld Devices*, October 2009
- [10] **Γεώργιος Β. Σομπόνης**, Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών. *Ευφυής Προσδιορισμός Βέλτιστων Διαδρομών βάσει Μεθόδων Μηχανικής Μάθησης*, Ιούλιος 2012
- [11] **Walter V, Kada M and Chen H**, Institute of Photogrammetry, Stuttgart University. *Shortest Path Analyses in Raster Maps for Pedestrian Navigation in Location Based Systems*
- [12] **Peter Sanders and Dominik Schultes**, University of Karlsruhe. *Engineering Fast Route Planning Algorithms*